

UNIVERSIDAD DE CUENCA



FACULTAD DE INGENIERÍA

ESCUELA DE INFORMÁTICA

**“Aplicación de Tecnologías Semánticas y Técnicas de Reconocimiento
de Objetos para la Identificación de Armas de Fuego en Video”**

UNIVERSIDAD DE CUENCA
desde 1867

Tesis previa a la obtención
del título de Ingeniero de Sistemas

Autores:

Edisson Geovanny Campoverde Llanos.
Alina Marisol Guerrero Fernández de Córdova.

Director:

Ing. Jorge Mauricio Espinoza Mejía, PhD.

Cuenca - Ecuador
2015

Resumen

Palabras clave: Televisión Digital, Reconocimiento de Objetos, Tecnologías Semánticas, Anotación Semántica.

La implementación de la Televisión Digital Terrestre en el Ecuador ha abierto un amplio campo de acción para proyectos e investigaciones relacionados a este tema. Dentro de este contexto, el análisis y clasificación del contenido violento dentro del ámbito televisivo se convierte en una tarea fundamental para lograr una presentación adecuada de la programación a los televidentes. En este trabajo se propone el desarrollo de un proceso para la anotación semántica de armas de fuego reconocidas en videos, con el propósito de contar con un repositorio semántico de la información recolectada para su posterior consulta. Para ello, se desarrolló en primer lugar un prototipo que, por medio de la aplicación de algoritmos, realiza la detección y el reconocimiento de armas de fuego en video. Por otro lado, se generó una ontología para modelar armas de fuego y su clasificación utilizando la metodología NeOn, la misma que se fusionó con una ontología que modela contenido televisivo. Posteriormente, se desarrolló un segundo prototipo en donde los datos de las armas de fuego reconocidas se anotaron semánticamente utilizando la ontología generada y se almacenaron en un repositorio diseñado para dicho fin. Finalmente, un tercer prototipo permitió a los usuarios realizar consultas al repositorio semántico acerca de la información almacenada. Con el afán de contribuir a la temática abordada se propone un proceso consolidado que engloba todo el trabajo realizado y que sirva de guía para proyectos que impliquen el reconocimiento de objetos y el uso de tecnologías semánticas para la anotación de los mismos.

*Edisson Geovanny Campoverde Llanos
Alina Marisol Guerrero Fernández de Córdova*

Abstract

Keywords: Digital Television , Object Recognition , Semantic Technologies, Semantic Annotation.

The implementation of the Digital Terrestrial Television in Ecuador has opened a wide field of action for projects and investigations related to this topic. From this context, the analysis and classification of televisual violent content becomes a fundamental task to accomplish in order to achieve a proper television programmes presentation to the audience. This project proposes the development of a process for the semantic annotation of firearms recognized in videos, with the purpose of generating a semantic repository containing the collected information so that it could be queried afterwards. To do this, a first prototype was developed which allows the detection and recognition of firearms appearing in video. Furthermore, an ontology was generated to model firearms and its classification by using NeOn methodology. This ontology was later fused with another ontology which models televisual content. Subsequently, a second prototype was developed to take the recognized firearms data, semantically annotate it using the generated ontology and later store it in a repository. Finally, a third prototype allowed the users to query the semantic repository. With the aim of contributing the addressed area of knowledge, a consolidated process is proposed. This process includes the whole executed work and serves as a guide for other projects implying objects recognition and the application of semantic technologies for their annotation.

*Edisson Geovanny Campoverde Llanos
Alina Marisol Guerrero Fernández de Córdova*

Índice general

Resumen	2
Abstract	3
Índice general	4
Índice de figuras.	7
Índice de tablas.	9
Índice de segmentos de código.	10
1. Introducción	18
1.1. Planteamiento del problema	18
1.2. Antecedentes	19
1.3. Justificación	20
1.4. Objetivos	21
1.4.1. Objetivo General	21
1.4.2. Objetivos Específicos	21
1.5. Alcance	21
1.6. Organización de la tesis	22
2. Contexto Tecnológico	24
2.1. Reconocimiento de objetos	24
2.1.1. Imagen	24
2.1.2. Video	25
2.1.3. Procesamiento de imágenes y video	25
2.1.4. Niveles de procesamiento de imágenes y video	26
2.1.5. Visión computacional	27
2.1.6. Reconocimiento de objetos en la actualidad	28
2.2. Algoritmos utilizados para el reconocimiento de objetos	30
2.2.1. Algoritmo de Análisis de Forma	30
2.2.2. Meanshift	30

2.2.3.	Camshift	31
2.2.4.	SIFT	32
2.2.5.	SURF	32
2.2.6.	FLANN	35
2.3.	Modelamiento de material multimedia mediante tecnologías semánticas	36
2.3.1.	Ontologías	36
2.3.2.	Lenguajes para la representación de ontologías	38
2.3.3.	Repositorios semánticos	39
2.3.4.	SPARQL	39
2.4.	Anotación semántica	39
2.4.1.	Semántica ontológica	40
2.4.2.	Concepto y utilidad	40
2.4.3.	Herramientas	40
2.5.	Metodologías para la creación de ontologías	41
3.	Reconocimiento de Objetos en Video	43
3.1.	Selección del algoritmo para el proceso de reconocimiento de objetos	43
3.2.	Selección de la herramienta para el desarrollo del módulo de reconocimiento de objetos	45
3.3.	Desarrollo de un prototipo para el reconocimiento de armas en video	48
3.3.1.	Entradas y salidas	50
3.3.2.	Herramientas de desarrollo	51
3.3.3.	Desarrollo	52
3.4.	Resultados	53
4.	Representación Formal de Objetos	57
4.1.	Recursos ontológicos y no ontológicos	58
4.2.	Especificación de requerimientos de la ontología	59
4.3.	Modelado de la ontología de armas de fuego.	63
4.4.	Desarrollo de un prototipo para la población de la ontología	68
4.4.1.	Entradas y salidas	69
4.4.2.	Herramientas de desarrollo	69
4.4.3.	Desarrollo	70



5. Anotación Semántica de Contenido Televisivo	74
5.1. Modelado formal para la anotación semántica de videos	74
5.2. Desarrollo de un prototipo para la anotación semántica de videos	78
5.2.1. Entradas y salidas	79
5.2.2. Requisitos previos a la programación	79
5.2.3. Desarrollo	79
5.3. Desarrollo de un prototipo para la consulta a un repositorio semántico	81
5.3.1. Entradas y salidas	82
5.3.2. Herramientas de desarrollo	83
5.3.3. Desarrollo	84
5.4. Proceso propuesto para reconocimiento y anotación semántica de objetos en video	87
5.4.1. Prototipos propuestos	88
5.4.2. Fases propuestas	90
6. Conclusiones y Futuras Líneas de Investigación	93
6.1. Conclusiones	93
6.2. Futuras Líneas de Investigación	96
Anexos	98
A. Configuración de Variables de Entorno para OpenCV	99
B. Ejecución del Módulo de Reconocimiento de Objetos	102
Acrónimos	107
Bibliografía.	108

Índice de figuras.

1.1.	Distribución de paquetes de trabajo del proyecto [1].	19
2.1.	Proceso de visión de alto nivel [2].	29
2.2.	Estructura de un sistema de visión basado en modelos [2].	29
2.3.	Reconocimiento mediante el algoritmo Meanshift [3].	31
2.4.	Reconocimiento mediante el algoritmo Camshift [3]. .	31
2.5.	Forma de onda Haar [4].	33
2.6.	Forma de onda Haar, se computan las respuestas en las direcciones x, y . El área oscura restará una unidad, el área más clara sumará una unidad [5].	33
2.7.	Puntos clave en una imagen usando SURF [6].	34
2.8.	Emparejamiento de puntos clave entre dos imágenes [6].	35
2.9.	Representación de una tripleta	37
3.1.	Esquema del módulo de reconocimiento.	49
4.1.	Categorización de los Recursos No Ontológicos [7]. . .	59
4.2.	Modelo ontológico de clasificación de las armas de fuego.	72
4.3.	Resultado de la ejecución del módulo de población . .	73
5.1.	Ontología Programmes [8].	77
5.2.	Metamodelo de la Ontología para anotación de videos.	78
5.3.	Resultados del emparejamiento de armas detectadas y registros en la ontología	80
5.4.	En rojo, Módulo de Consulta al Repositorio Semántico	82
5.5.	Captura de la interfaz Inicio	85



5.6.	Captura de la interfaz Consultas Predefinidas	86
5.7.	Captura de la interfaz Consultas Personalizadas . . .	87
5.8.	Captura de la interfaz SPARQL endpoint	88
5.9.	Esquema del proceso de reconocimiento y anotación de objetos en video	89
A.1.	Ventana de configuración avanzada del sistema. . . .	99
A.2.	Ventana de propiedades del sistema.	100
A.3.	Ventana de configuración de variables de entorno. . .	101
B.1.	Interfaz del módulo de reconocimiento de armas en video.	102
B.2.	Imagen representativa de un video en reproducción. .	103
B.3.	Imagen representativa del reconocimiento exitoso de tres objetos simultáneamente.	104

Índice de tablas.

2.1. Análisis Comparativo de las Metodologías.	42
3.1. Análisis Comparativo de los algoritmos para reconocimiento de video.	45
3.2. Herramientas para el análisis de imágenes y video . .	47
3.3. Tabla comparativa de herramientas para la visión computacional.	48
3.4. Tabla de resultados del módulo de las pruebas realizadas.	55
3.5. Porcentaje de efectividad de reconocimiento.	56
4.1. Documento de Especificación de Requisitos: Secciones 1 a 5.	60
4.2. Documento de Especificación de Requisitos: Sección 6.	61
4.3. Documento de Especificación de Requisitos - Sección 7.	62
4.4. Ontologías relacionadas con armas de fuego.	64
4.5. Cuadro comparativo de Ontologías.	65
4.6. Tabla de calificación de ontologías candidatas.	66
5.1. Selección de ontología a reutilizar.	76

Índice de segmentos de código.

3.1. Estructura del archivo .xml	51
4.1. Consulta SPARQL para obtener las armas almacenadas en DBpedia.	71
5.1. Formato utilizado para registro de escenas violentas. .	81
B.1. Contenido del archivo .xml obtenido como salida. . .	104



Yo, *Edisson Geovanny Campoverde Llanos*, autor de la tesis *Aplicación de Tecnologías Semánticas y Técnicas de Reconocimiento de Objetos para la Identificación de Armas de Fuego en Video*, certifico que todas las ideas, opiniones, y contenidos expuestos en la presente investigación, son de exclusiva responsabilidad de sus autores.

Cuenca, Septiembre 2015.

UNIVERSIDAD DE CUENCA
desde 1867

Edisson Geovanny Campoverde Llanos
C.I. 0104993811



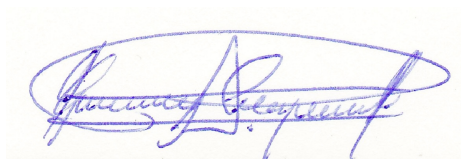
Yo, *Alina Marisol Guerrero Fernández de Córdova*, autor de la tesis *Aplicación de Tecnologías Semánticas y Técnicas de Reconocimiento de Objetos para la Identificación de Armas de Fuego en Video*, certifico que todas las ideas, opiniones, y contenidos expuestos en la presente investigación, son de exclusiva responsabilidad de sus autores.

Cuenca, Septiembre 2015.

Alina Marisol Guerrero Fernández de Córdova
C.I. 0104641741

Yo, *Edisson Geovanny Campoverde Llanos*, autor de la tesis *Aplicación de Tecnologías Semánticas y Técnicas de Reconocimiento de Objetos para la Identificación de Armas de Fuego en Video*, reconozco y acepto el derecho de la Universidad de Cuenca, en base al Art. 5 literal c) de su Reglamento de Propiedad Intelectual, de publicar este trabajo por cualquier medio conocido o por conocer, al ser este requisito para la obtención de mi título de *Ingeniero de Sistemas*. El uso que la Universidad de Cuenca hiciere de este trabajo, no implicará afección alguna de mis derechos morales o patrimoniales como autor.

Cuenca, Septiembre 2015.



Edisson Geovanny Campoverde Llanos
C.I. 0104993811



Yo, *Alina Marisol Guerrero Fernández de Córdova*, autor de la tesis *Aplicación de Tecnologías Semánticas y Técnicas de Reconocimiento de Objetos para la Identificación de Armas de Fuego en Video*, reconozco y acepto el derecho de la Universidad de Cuenca, en base al Art. 5 literal c) de su Reglamento de Propiedad Intelectual, de publicar este trabajo por cualquier medio conocido o por conocer, al ser este requisito para la obtención de mi título de *Ingeniero de Sistemas*. El uso que la Universidad de Cuenca hiciere de este trabajo, no implicará afección alguna de mis derechos morales o patrimoniales como autor.

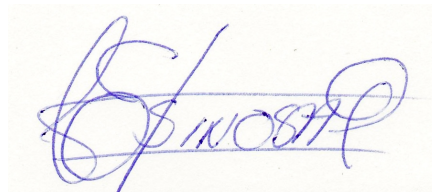
Cuenca, Septiembre 2015.

Alina Marisol Guerrero Fernández de Córdova
C.I. 0104641741

CERTIFICO

Que el presente proyecto de tesis: Aplicación de Tecnologías Semánticas y Técnicas de Reconocimiento de Objetos para la Identificación de Armas de Fuego en Video fue dirigido por mi persona. _____

UNIVERSIDAD DE CUENCA
desde 1867



Ing. Jorge Mauricio Espinoza Mejía, PhD.
Director de Tesis

Agradecimientos

En primer lugar queremos agradecer al Ing. Mauricio Espinoza Mejía PhD, nuestro tutor, por habernos brindado su apoyo incondicional, sus valiosos conocimientos y prácticos consejos durante todo el desarrollo de este proyecto. A nuestra alma máter, la Universidad de Cuenca, a nuestra querida Facultad de Ingeniería y a todos los profesores que con su esfuerzo y dedicación contribuyeron día a día en nuestra formación académica. A nuestros familiares y amigos por todo el tiempo, apoyo y cariño brindado.

A todos y todas gracias de todo corazón.

Dedicatoria

A mis padres, hermanos, familiares y amigos por haber sido el apoyo fundamental durante este largo pero satisfactorio camino lleno de sinsabores, alegrías y triunfos. A mi abuela, que donde sea que ella se encuentre, ha guiado y seguirá guiando mis pasos cada día. Gracias a todos infinitamente.

Edisson Geovanny Campoverde Llanos.


UNIVERSIDAD DE CUENCA
desde 1867

A todos aquellos que fueron mi apoyo incondicional durante mis años de estudio y que durante el desarrollo de este proyecto me brindaron su aliento, su sabiduría y su buen ánimo para concluir con éxito.

Alina Marisol Guerrero Fernández de Córdova.

Capítulo 1

Introducción

1.1. Planteamiento del problema

El procesamiento de video puede definirse como la captura, manipulación, almacenamiento y transmisión de un conjunto de imágenes que representan escenas en movimiento y que pueden ser presentadas a través de una pantalla. Partiendo desde el estándar D1 [8] introducido en el año 1987 por Sony¹; mismo que es considerado uno de los primeros formatos utilizados en el ámbito televisivo, el procesamiento de video ha experimentado un desarrollo continuo hasta los estándares de avanzada que se tiene hoy en día como es el caso de *Moving Pictures Experts Group 2* (MPEG-2) y *Moving Pictures Experts Group 4* (MPEG-4) [9] aplicados en el estándar japonés-brasileño *Integrated Services Digital Broadcasting - Terrestrial* (ISDB-T) ² para la televisión digital, el mismo que está siendo implementando en nuestro país.

Actualmente todo video contiene metadatos, los cuales representan información relacionada a su formato, tamaño, duración, dimensiones, fecha de creación, entre otros. Dicha información es utilizada por los reproductores de video para procesar y presentar el video de manera adecuada en los diferentes dispositivos. Sin embargo, existe un problema debido a que los metadatos que permiten especificar un video de la televisión digital, no contienen información específica referente al contenido violento que puedan presentar. Entiéndase por información específica, a metadatos que describan cuál es el contenido violento y su intervalo de aparición. Hoy en día se realiza un proceso manual de categorización de contenido, llevado a cabo por expertos en el tema, quienes a su juicio y en base a ciertos estándares, determinan si un contenido es o no adecuado para un tipo

¹<http://www.sony.net>

²<http://www.dibeg.org/>

de público específico. En el caso de Ecuador, este proceso se lleva a cabo por el Consejo de Regulación y Desarrollo de la Información y Comunicación [10].

En base a lo dicho anteriormente, surge la necesidad de contar con un sistema que permita analizar el contenido de un video. Esto con el objetivo de obtener información que facilite el enriquecimiento de los metadatos actuales, lo cual será la base para el desarrollo de un mecanismo para la automatización del proceso de clasificación de contenido televisivo.

Debido a la variedad de contenido televisivo existente, una de las necesidades más básicas consiste en la determinación de si una escena es violenta o no. Para este proyecto de tesis se ha elegido el tema de violencia para realizar una prueba de conceptos. Concretamente, los elementos elegidos son las armas de fuego, pues su aparición determina la presencia de violencia. Es por ello que contar con un sistema que permita detectar y reconocer estos elementos para luego modelar formalmente dicha información se puede considerar una buena contribución para el problema planteado.

1.2. Antecedentes

El creciente desarrollo de la televisión digital en el país ha abierto nuevos campos de investigación. Por tal motivo el Departamento de Ciencias de la Computación, conjuntamente con investigadores y desarrolladores, han incursionado en nuevos proyectos relacionados a este tema. Uno de ellos es el denominado “Empleo de tecnologías semánticas para el análisis de contenido multimedia transmitido para televisión digital terrestre” [1], cuyo objetivo es desarrollar soluciones innovadoras que permitan afrontar los retos en cuanto a la gestión de contenidos y la recepción de la señal de televisión digital.

Dicho proyecto contempla cuatro ejes o paquetes de trabajo, cada uno con un objetivo específico y que presentará como resultado un prototipo, tal y como se puede observar en la Figura 1.1.

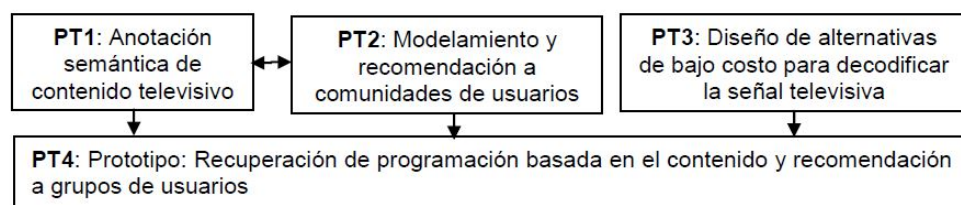


Figura 1.1: Distribución de paquetes de trabajo del proyecto [1].

El primer paquete contempla la personalización de contenidos mediante la anotación semántica del audio y video. Entiéndase por semántica al significado que se le da a las entidades o conceptos del mundo real y que forman parte de una estructura definida formalmente [11]. El propósito es enriquecer los metadatos agregando servicios de valor añadido al usuario; siendo esto el punto de partida para este proyecto de tesis, el cual consiste en brindar una solución para el problema planteado centrándose en el análisis de videos para la detección y anotación de armas de fuego presentes en los mismos.

1.3. Justificación

Para el territorio ecuatoriano en el Artículo 65 de la Ley Orgánica de Comunicación se presenta la clasificación de contenido televisivo en tres categorías: A (Apto para todo público), B (Apto para todo público con vigilancia de una persona adulta), C (Apto para personas adultas) [10]. De la misma manera cada país maneja una clasificación propia en base a lo que estipulen sus leyes. Esta es la denominada clasificación por edades, la cual a su vez dictamina el horario en el cual debe transmitirse cada tipo de programación televisiva.

Esta manera de clasificar contenidos se basa únicamente en el juicio de los expertos siguiendo criterios de calificación como violencia, conflictividad, sexo, lenguaje inapropiado, entre otros; criterios que presentan pequeñas variaciones de acuerdo a cada país.

Como se puede evidenciar, el contenido televisivo es muy amplio por lo cual en un primer acercamiento el sistema que se propone desarrollar cubrirá la detección de armas de fuego en una escena, permitiendo así catalogar a un video como violento por el hecho de contener una arma en el algún/algunos fragmento(s) del mismo.

Al automatizar esta tarea los beneficiarios directos serán los medios televisivos los cuales podrán contar con una fuente de información acerca de su programación que permita catalogar de mejor manera la programación transmitida a sus televidentes.

Si bien el contenido televisivo necesita clasificación, la herramienta propuesta puede ser aplicada para anotar otro tipo de recursos de video como lo son películas, series, episodios, etc.; que se pueden encontrar en la web.

1.4. Objetivos

1.4.1. Objetivo General

Desarrollar un proceso de anotación semántica basado en la información proveniente de técnicas de reconocimiento de objetos utilizadas para identificar la presencia de armas de fuego en secuencias de video.

1.4.2. Objetivos Específicos

- Realizar el reconocimiento de un conjunto determinado de armas de fuego en un video específico mediante la generación de un prototipo.
- Generar una ontología de armas de fuego siguiendo una metodología específica.
- Fusionar la ontología de armas de fuego con una ontología para la modelación de programas de televisión dando como resultado una ontología que permita la anotación de armas de fuego en video.
- Anotar semánticamente un video en base a los términos de la ontología para la posterior búsqueda y/o consulta de resultados y generación de los prototipos correspondientes.

1.5. Alcance

El objetivo principal que se plantea en el presente proyecto de tesis, es el desarrollo de un proceso para la anotación semántica de videos, dicho proceso será dividido en diferentes fases las cuales se describen a continuación:

La fase inicial consistirá en realizar el reconocimiento de armas de fuego en un video que se recibe como fuente de entrada; para ello se aplicarán algoritmos de detección y reconocimiento de objetos, de donde se obtendrá como resultado un archivo con la información recolectada de dicho proceso incluyendo el instante en el que aparecen en el video.

La siguiente fase se centrará en generar una ontología de armas de fuego siguiendo una metodología específica y en base a información recolectada de sitios confiables acerca de la clasificación de las diferentes armas de fuego. Esto con el fin de poder relacionar los objetos detectados en el video con las instancias

presentes en la ontología; para ello se codificará un prototipo que permita verificar las ocurrencias en base a un coeficiente de aproximación. Una vez hecho esto, se fusionará la ontología generada con una ontología que permita modelar programación televisiva, lo cual permitirá realizar la anotación semántica de los videos.

Los resultados obtenidos de la anotación serán colocados en un repositorio semántico al cual se le podrán realizar consultas a través de un módulo que se generará como fase final de este proyecto.

Cada fase de este proyecto se realizará en módulos tan independientes como sea posible pensando en su futura reutilización en otras áreas de estudio relacionadas a las tecnologías semánticas, detección y reconocimiento de objetos o cualquier otro proyecto pertinente. El resultado final, será la documentación obtenida del desarrollo de este proyecto, de manera que constituya una contribución útil y que pueda ser vista como una herramienta fácilmente aplicable en el ámbito informático.

1.6. Organización de la tesis

El presente proyecto de tesis tendrá la siguiente estructura:

- **Capítulo 1: Introducción.** Presentación del proyecto de tesis incluyendo su problemática, antecedentes, objetivos, justificación y alcance.
- **Capítulo 2: Contexto Tecnológico.** Introducción a los conceptos relacionados con los ámbitos incluidos en este proyecto de tesis tales como detección y reconocimiento de objetos en video, metodologías para la generación de ontologías y anotación semántica.
- **Capítulo 3: Reconocimiento de Objetos en Video.** Selección y aplicación de algoritmos para el reconocimiento de armas de fuego en un video. Se producirá como resultado un prototipo que realice dicha tarea.
- **Capítulo 4: Representación Formal de Objetos.** Creación de una ontología que modele armas de fuego y su clasificación en base a una metodología específica y población de la misma con datos extraídos de la web.
- **Capítulo 5: Anotación Semántica de Contenido Televisivo.** Desarrollo de un prototipo para la anotación semántica de armas de fuego en video. Construcción de un repositorio donde se alojen los resultados para la

realización de consultas por medio de un prototipo creado para dicho fin. Finalmente, consolidación de un proceso para proyectos que impliquen detección y reconocimiento de objetos en video así como la anotación semántica de los mismos.

- **Capítulo 6: Conclusiones y Futuras líneas de investigación.** Presentación de conclusiones generales y futuras líneas de investigación.
- **Anexos.** Documentos adjuntos.
- **Acrónimos.** Listado de acrónimos utilizados en este documento.
- **Bibliografía.** Referencias y fuentes bibliográficas de consulta e información.



UNIVERSIDAD DE CUENCA
desde 1867

Capítulo 2

Contexto Tecnológico

En este capítulo se abordarán, desde un contexto teórico, temas como el reconocimiento de objetos, modelamiento de material multimedia mediante tecnologías semánticas, anotación semántica y finalmente, metodologías para la creación de ontologías. Estos temas servirán como base de conocimiento para el desarrollo de los siguientes capítulos de esta tesis.

2.1. Reconocimiento de objetos

El reconocimiento de objetos es una rama del procesamiento de imágenes y video que se encuentra en constante desarrollo actualmente. Para este proyecto de tesis es necesario desarrollar un módulo que permita reconocer objetos en un video. Para efectos de implementación de un prototipo, se desarrollará un módulo que permita reconocer armas de fuego en un video específico, que será fraccionado en imágenes digitales con el fin de aplicar algoritmos de reconocimiento y producir así una salida con información acerca de los objetos encontrados.

El campo del reconocimiento de objetos es amplio y puede ser aplicado en diferentes ámbitos. A continuación se introducirán conceptos referentes al tema y pertinentes al desarrollo de esta tesis.

2.1.1. Imagen

Según la *Real Academia de la Lengua Española* (RAE)¹, imagen es la figura, representación, semejanza y apariencia de algo. Para el ámbito computacional, una imagen puede ser concebida como la representación visual de cosas en forma digital, a esto se le llama imagen digital[12]. Una imagen digital puede estar repre-

¹<http://www.rae.es/>



sentada por píxeles o por vectores y el archivo que almacena el computador puede tener diferentes formatos de acuerdo a la calidad, dimensiones y/o características que se almacenen, así como el uso que se le dé a dicha imagen.

2.1.2. Video

Según la RAE, video es un sistema de grabación y reproducción de imágenes, acompañadas o no de sonidos. Este concepto varía según el ámbito en el que se utilice pero conservando su característica principal de representar un conjunto de imágenes. En el ámbito computacional, se puede describir al video digital como un conjunto de cuadros compuestos por píxeles, grabados o reproducidos con un intervalo de tiempo pequeño, de manera que la transición entre imagen e imagen sea imperceptible al ojo humano. Así como las imágenes, el video digital se puede almacenar en diferentes formatos.

2.1.3. Procesamiento de imágenes y video

Durante el transcurso de la historia, el ser humano siempre ha buscado interpretar todo lo que ve, y la manera más básica de interpretación se realiza mediante sentido de la visión, que es uno de los sentidos más desarrollados en los seres humanos. Una persona observa algo y lo interpreta en su corteza cerebral dando el significado correcto y acertado de lo observado. La visión permite a la persona percibir su entorno desde el momento en el que nace. Le permite captar colores, formas, profundidad y cualquier otra característica; que asociada a la percepción de los demás sentidos, permite al humano tomar decisiones en la vida cotidiana. Este proceso es mecánico y el humano lo hace sin esfuerzo alguno. Pero, ¿qué sucede si se requiere que una máquina imite este comportamiento?

Esta misma tarea no es tan fácil para un computador que trabaja de manera digital. Un computador por sí solo no comprenderá el contenido de un video. El simple hecho de captar video con una cámara y presentarlo en pantalla implica procesamiento de video en el computador, pero esto no significa procesamiento de su contenido, de esto se hablará más adelante.

Procesamiento de imágenes se entiende como el hecho de realizar ciertas tareas sobre una fuente de entrada (imagen) con el fin de mejorar o modificar sus características para adecuarla a una aplicación específica. Según algunos autores [13] [14], el procesamiento de imágenes puede limitarse al proceso que comprende operaciones cuyo origen es una imagen y cuyo resultado final es otra imagen. Sin

embargo esto ha causado controversia; pues de tomarse como cierto, cualquier operación que implique extracción de información ya no formaría parte de este concepto y pasaría a formar parte de lo que se conoce como visión computacional, tratado en la Sección 2.1.5

Tomando como base dicha definición el procesamiento digital de video consiste en procesar un conjunto de imágenes aplicando diferentes técnicas con el objetivo de mejorar o modificar sus características.

2.1.4. Niveles de procesamiento de imágenes y video

Según [14], para tener un entendimiento más acertado acerca del procesamiento de imágenes se puede decir que este sigue un paradigma dividido en tres niveles:

- Nivel Bajo
- Nivel Intermedio
- Nivel Alto

El proceso de nivel bajo se encuentra relacionado básicamente con las operaciones encargadas de dar tratamiento a las imágenes como la eliminación de ruido, mejoramiento de brillo y saturación, etc. En este nivel, tanto la entrada como la salida de información son imágenes.

El proceso de nivel intermedio está relacionado con las operaciones encargadas de segmentación de regiones y/u objetos para así poder clasificarlos y describirlos en conjuntos de objetos o de manera individual. En este nivel la entrada es una imagen y la salida corresponde a los atributos obtenidos a partir de esta. Dichos atributos pueden ir desde la identificación de contornos y esquinas hasta determinar objetos por completo.

El proceso de nivel alto tiene como objetivo fundamental dar significado a los atributos obtenidos en el nivel intermedio, con el fin de relacionarlos con funciones cognitivas asociadas a la visión, tratando así de emular la visión humana para poder inferir situaciones de la vida real. En este nivel la entrada son los atributos obtenidos a partir del procesamiento de una imagen y como salida se obtiene el “significado” de los atributos.

Tanto el nivel alto como el nivel intermedio pueden considerarse como visión computacional.

2.1.5. Visión computacional

Como se mencionó en la Sección 2.1.4, visión es la capacidad de los seres vivos para percibir el entorno a través de la vista y procesar mediante el cerebro la información recolectada. La Visión Computacional o *Computer Vision* en inglés, emula esta capacidad teniendo como propósito que un computador pueda reconocer y procesar características de imágenes o videos proporcionados como entrada; para producir una salida que sirva a los intereses de los usuarios del sistema. Entre las tareas principales de la Visión Computacional se encuentran la detección, segmentación, localización y seguimiento de objetos o áreas específicas de las imágenes [14] [15] [2].

Gracias al incremento de la capacidad de procesamiento de los computadores, así como a los avances realizados en el procesamiento de imagen y video; hoy en día la visión computacional está en pleno desarrollo. Diferentes campos de la industria encuentran en la visión computacional un campo de explotación para sus intereses específicos. La medicina, la milicia, la astronomía, la ingeniería, la policía, compañías de vigilancia, incluso el marketing utilizan detección, reconocimiento y seguimiento de objetos y personas para varios fines, obteniendo cada vez mejores resultados [16].

Dentro de la visión computacional, el reconocimiento de objetos está catalogado como procesamiento de imagen a nivel intermedio. En este ámbito se tienen tres corrientes principales: la segmentación, detección de contornos y detección de regiones [2] .

La segmentación busca características similares de los píxeles como: intensidad, textura, color entre otras; para en base a ello, determinar un objeto en la imagen. Existen varias técnicas para la segmentación, basadas en grafos [17], espacios de estados, árboles e incluso semántica de dominios [2]. Más detalle acerca de este tema puede encontrarse en el trabajo de reconocimiento de regiones presentado en [18].

La detección de contornos y de regiones están íntimamente ligadas, ambas se basan en hallar las características similares de los objetos en una imagen. Encontrar una región permite determinar el borde que divide una región con las vecinas. Encontrar los bordes es esencial para delimitar regiones y viceversa [2]. Debido a la disponibilidad de algoritmos para detección de contornos, regiones y puntos clave, así como a la abundante documentación, se eligió este enfoque para aplicarlo en el desarrollo del módulo de reconocimiento.

Por otro lado, considerando que el mundo es una representación continua, pero

el video digital es una secuencia de imágenes que capturan escenas estáticas, se han planteado maneras distintas para manejar el video en la visión computacional. Se puede elegir entre un enfoque continuo o discreto [2].

El enfoque continuo considera la secuencia de imágenes como un flujo de píxeles en los cuales se dan variaciones de intensidad y color denominado flujo óptico. El enfoque discreto por el contrario, toma en cuenta la secuencia como tal, como un conjunto de imágenes estáticas que serán tratadas de forma independiente o por detección de variaciones en las mismas. Es este segundo enfoque el que será utilizado para el desarrollo de este proyecto de tesis.

2.1.6. Reconocimiento de objetos en la actualidad

En la actualidad las tareas de reconocimiento de objetos tienen un fin que va más allá de la mera aplicación de algoritmos para detectar la presencia de cierto objeto en video, más bien se trata de reconocer objetos para luego analizar lo encontrado, generar información, almacenarla y analizarla para producir conocimiento. Por lo que se puede hablar ya de visión computacional de alto nivel [14].

Como se mencionó, la visión computacional de alto nivel pretende interpretar y dar significado a los datos obtenidos mediante la aplicación de diferentes algoritmos y técnicas. Por ejemplo, el proceso que se ilustra en la Figura 2.1 requiere una representación interna que describe los objetos en el dominio de interés, para luego hacer una correspondencia por inferencia con la información de entrada, produciendo finalmente una salida. En un inicio se tendrán las definiciones de las distintas características, las cuales se buscarán en una imagen y se obtendrá dicha imagen segmentada de acuerdo a las características. Luego, usando tanto las características como la imagen segmentada, se procederá a comparar con la entrada que se obtiene del mundo real (representación del mundo), por ejemplo un nuevo video, y finalmente se obtendrá una salida de lo que se ha detectado en un formato definido por los desarrolladores (descripción simbólica). A esto se le conoce como segmentación semántica [2]. Una aplicación de esto puede ser encontrada en [19] y [18] donde se toman en cuenta las características de los píxeles para dividir una imagen en segmentos e inferir el material y/u objeto al cual representan dichos píxeles.

Otro enfoque más sencillo pero no menos utilizado es el enfoque basado en modelos, en donde se detectan contornos o regiones de imágenes modelo para realizar correspondencia con imágenes del mundo real. Como se puede observar

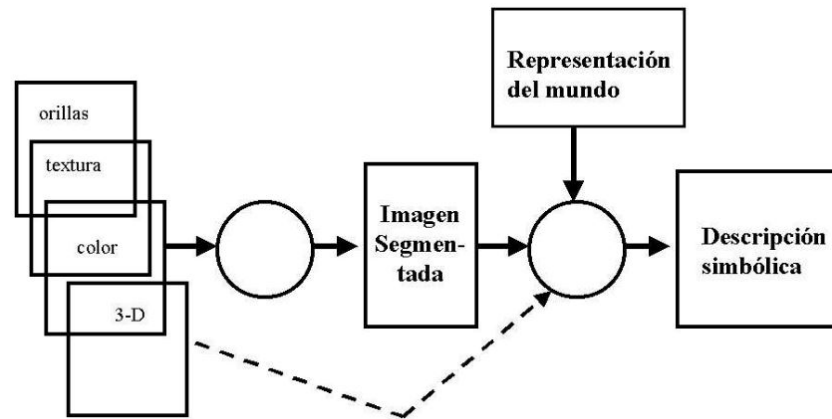


Figura 2.1: Proceso de visión de alto nivel [2].

en la Figura 2.2, el proceso inicia tomando como base al mundo real y realizando el modelado de los objetos a reconocer. Así mismo se obtendrán imágenes que representan al mundo real, de las cuales se extraerán características y se obtendrá una descripción de cada imagen. La descripción de la imagen se comparará con el modelo existente y se determinará por reconocido el objeto cuando el porcentaje de similitud sea lo suficientemente alto.

Este último es ampliamente utilizado en trabajos que implican reconocimiento de objetos, cuenta con extensa documentación y su desarrollo es menos complejo que la segmentación semántica. Debido a estas razones, se ha elegido este enfoque para implementar un prototipo que realice la detección de armas de fuego en base a imágenes previamente recolectadas.

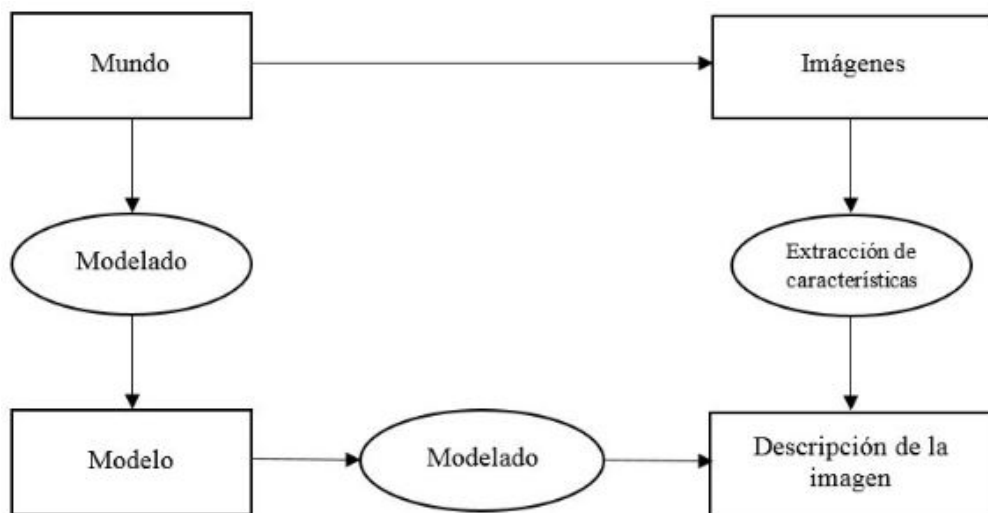


Figura 2.2: Estructura de un sistema de visión basado en modelos [2].

2.2. Algoritmos utilizados para el reconocimiento de objetos

Para desarrollar un sistema que reconozca objetos en video, es necesario conocer los algoritmos que se pueden utilizar para dicho fin. Se presenta a continuación una breve descripción de algunos algoritmos que pueden ser de utilidad en este proyecto. El primer algoritmo representa un enfoque mixto entre modelos y segmentación semántica, mientras que los algoritmos restantes aplican un enfoque orientado a modelos. Posteriormente, en el Capítulo III, se determinará cuál es el algoritmo más adecuado para implementar el prototipo de reconocimiento de armas de fuego.

2.2.1. Algoritmo de Análisis de Forma

El *Algoritmo de Análisis de Forma* (AFA) es un algoritmo que permite determinar uno o varios objetos en movimiento dentro de un video [20]. Como dato de entrada recibe una imagen bitonal (blanco y negro), la misma que pasa por un proceso de tratamiento de píxeles en donde se analizan regiones de los mismos buscando aquellos píxeles conexos y convexos para así poder determinar el objeto u objetos que se están moviendo. Un píxel conexo hace referencia a un píxel cuya vecindad de píxeles tienen el mismo color, por lo tanto forman parte de la misma región. Por otro lado un píxel convexo es aquel que difiere de su vecindad, lo cual indica que forma parte de otra región que posiblemente sea la del objeto en movimiento. El AFA permite determinar las regiones dentro de un video, aunque es un proceso muy costoso y difícil de implementar ya que analiza fotograma por fotograma en búsqueda de los cambios que los píxeles presenten.

2.2.2. Meanshift

Meanshift es un algoritmo utilizado para encontrar y seguir objetos en videos. Su funcionamiento se basa en partir de un conjunto de puntos, por ejemplo un conjunto de píxeles de una imagen obtenida a través de retroproyección por histograma [21], mediante lo cual se obtiene una imagen bitonal. Luego se posiciona una ventana o área en la imagen; por ejemplo un círculo o un rectángulo, y se mueve dicha ventana hasta que coincida con el área de mayor densidad de píxeles. Para desplazar la ventana desde el área inicial hacia el área con mayor densidad se compara el centro de la ventana con el punto en donde hay mayor densidad de

píxeles. En caso de coincidir el proceso se detiene y el objeto está ubicado. Caso contrario, se desplaza el centro de la ventana hacia el punto encontrado, luego se repite el proceso hasta que el punto central y el punto más denso coincidan. La entrada del algoritmo es la imagen retroproyectada y un punto objetivo inicial. Cuando el objeto se mueve, el movimiento se refleja en la imagen y el algoritmo desplazará la ventana hasta la ubicación correcta como se observa en la Figura 2.3. Una desventaja es que el algoritmo trabaja bien solo con objetos que se diferencian claramente de su fondo. [3]



Figura 2.3: Reconocimiento mediante el algoritmo Meanshift [3].

2.2.3. Camshift

Este algoritmo recibe el nombre de *Continuously Adaptive Meanshift*. Básicamente es el algoritmo Meanshift pero tiene la capacidad de redimensionar la ventana conforme el objeto se mueve. Luego realiza la siguiente iteración tomando la nueva ventana como base [3]. Un ejemplo de ello se observa en la Figura 2.4.



Figura 2.4: Reconocimiento mediante el algoritmo Camshift [3].

2.2.4. SIFT

Así como Meanshift, otro algoritmo invariante a la escala es *Scale-Invariant Feature Transform* (SIFT), el cual se desarrolló en 2003 [22]. El objetivo del algoritmo es representar la imagen por medio de puntos de interés o keypoints que contienen información relevante de la imagen. Dichos puntos pueden reconocerse independientemente de las dimensiones de la imagen o de su orientación, es decir que el algoritmo es invariable a escala y además a orientación, lo cual lo hace más eficaz que los nombrados anteriormente. Cada punto de interés posee un descriptor que es básicamente una descripción robusta y única del área que rodea el punto clave de manera que pueda reconocerse la ocurrencia del mismo punto en cualquier otra imagen.

El algoritmo puede ser descrito en cuatro pasos:

1. Se representa la imagen en diferentes escalas y tamaños para poder identificar posibles puntos de interés que se presenten invariables en las distintas representaciones de la imagen.
2. Se localizan los puntos clave, que son aquellos que se mantienen independientemente de los cambios de escala.
3. Se asigna una dirección a cada punto clave de acuerdo un análisis del gradiente y la vecindad del punto.
4. Se calcula un descriptor para cada punto clave. De manera que al procesar una nueva imagen se pueda reconocer los puntos clave de una imagen procesada anteriormente y detectar si es una ocurrencia de la imagen anterior. En cuyo caso se dirá que se detectó el objeto presente en la primera imagen.

2.2.5. SURF

Publicado en 2006, el algoritmo *Speeded Up Robust Features* (SURF) [23], es un algoritmo basado en características, que de la misma manera que SIFT, se basa en la detección de puntos clave, invariables a escala y rotación. La diferencia se encuentra en la manera en que ambos algoritmos generan los descriptores para sus puntos de interés.

SURF basa la generación de descriptores en formas de onda Haar para lograr mayor eficiencia con respecto a SIFT. La forma de onda Haar, como se observa en la Figura 2.5, es una secuencia de funciones de forma cuadrada re-escaladas que

juntas forman una familia de formas de onda. Es considerada la forma de onda más básica cuyo análisis puede hacerse en base a una forma de onda ortonormal, más información puede ser encontrada en [24].

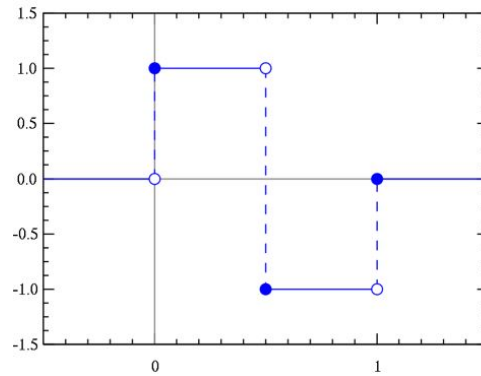


Figura 2.5: Forma de onda Haar [4].

El proceso de generación de descriptores consiste en dividir la imagen en regiones rectangulares, a su vez se dividen dichas regiones en sub-rectángulos y se usan formas de onda Haar debidamente orientadas en una región cuadrada alrededor de los puntos de interés para encontrar gradientes de intensidad en las direcciones XY, esto permitirá determinar el entorno que rodea a un punto de interés, para que sea posible detectar su ocurrencia en otras imágenes. Como se observa en la Figura 2.6, se dividen las regiones en sub-rectángulos, de acuerdo a la intensidad que presenten en la figura original valores +1 y -1 para luego realizar una sumatoria, determinando así los descriptores del punto de interés.

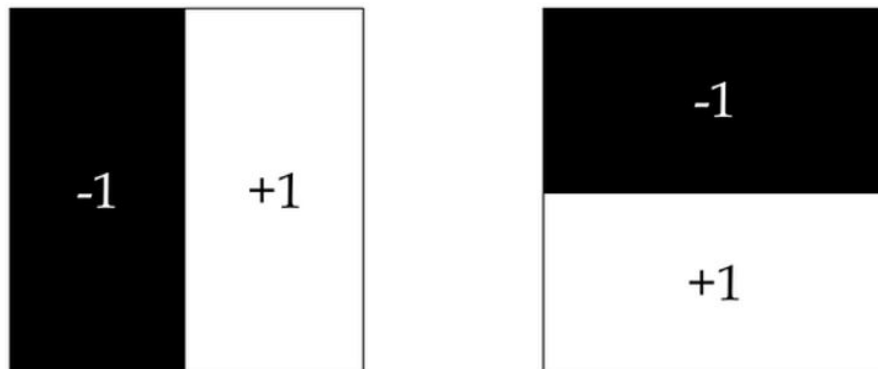


Figura 2.6: Forma de onda Haar, se computan las respuestas en las direcciones x, y . El área oscura restará una unidad, el área más clara sumará una unidad [5].

El algoritmo SURF inicia su proceso de reconocimiento mediante la declaración de un valor mínimo Hessiano que se utilizará para comparar con la matriz

Hessiana y determinar la validez de los puntos clave a detectarse. La matriz Hessiana es una matriz que se construye con las derivadas parciales de la intensidad de los píxeles de la imagen. El determinante de esta matriz se conoce como Hessiano que indica que tan robusto es un píxel para constituir el centro del punto clave. El valor mínimo Hessiano es una variable de frontera que decide desde qué valor se aceptarán los puntos clave. En la práctica, mientras más alto este valor, menor cantidad de puntos se obtendrán; pero estos serán más repetitivos y por lo tanto más útiles. Por otro lado, mientras menor sea dicho valor, más puntos se obtendrán pero también más ruido. El valor usual utilizado para imágenes oscila entre 400 y 800 [23].

Inicialmente el algoritmo recibe como entrada una imagen ya sea en formato *Portable Network Graphics* (PNG), *Joint Photographic Experts Group* (JPEG) u otro dependiendo del lenguaje de implementación. El algoritmo detecta los puntos clave en la imagen y devuelve un vector que contiene dichos puntos. Cada punto se encuentra descrito por sus coordenadas X - Y , el tamaño que representa el diámetro del área útil adyacente al punto detectado, el valor Hessiano del punto, un identificador, entre otros parámetros dependiendo del contexto de aplicación. Luego se procede a extraer los descriptores correspondientes a los puntos clave, para ello el algoritmo recibe como entrada la imagen y el vector de puntos clave generado en el proceso anterior, devolviendo como salida el conjunto de descriptores para dicha imagen. En base a los puntos clave y los descriptores detectados se pueden realizar varias tareas como graficación, emparejamiento y/o análisis de acuerdo a los requerimientos que se presenten.

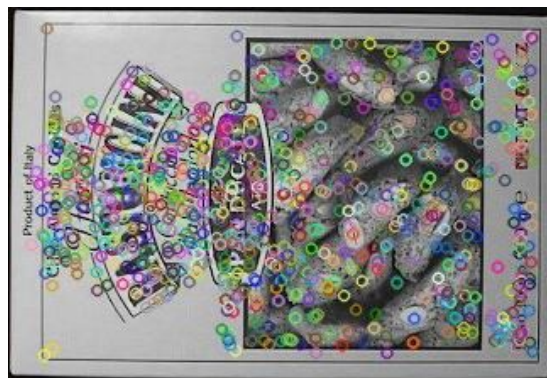


Figura 2.7: Puntos clave en una imagen usando SURF [6].

Como ejemplo de lo que puede lograr SURF, en la Figura 2.7 se muestran los puntos clave obtenidos en una imagen de entrada, cada punto está encerrado en un círculo. Estos puntos representan a la imagen de manera única y podrán

compararse posteriormente con otros conjuntos de puntos para determinar si el objeto está o no presente.

En la Figura 2.8 se muestra el resultado de detectar los puntos clave en dos imágenes y realizar el emparejamiento entre los dos conjuntos de puntos. Se puede observar que a pesar de la inclinación de la segunda imagen, los puntos clave han sido detectados y emparejados correctamente.

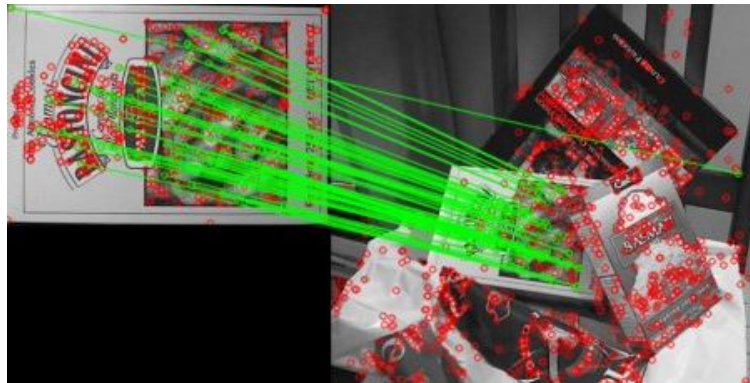


Figura 2.8: Emparejamiento de puntos clave entre dos imágenes [6].

2.2.6. FLANN

Fast Library for Approximate Nearest Neighbors (FLANN) es una librería que contiene una colección de algoritmos optimizados para la búsqueda rápida del vecino más cercano en grandes conjuntos de datos. En esta librería se proveen algoritmos basados en la investigación de D. Muja y D. Lowe [25] quienes se basan en usar algoritmos de árboles k-d y k-means; detallados en [17] y [26], con el objetivo de aumentar la eficiencia de los algoritmos de búsqueda del vecino más cercano, logrando una eficiencia de entre el 60 y 90 por ciento. Es por ello, que estas librerías se pueden aplicar eficientemente en combinación con otros algoritmos de reconocimiento.

Para tareas de procesamiento de imagen se utiliza la librería FLANN para emparejamiento de puntos clave y descriptores, es decir, para determinar la ocurrencia del mismo conjunto de descriptores en dos o más imágenes. Una aplicación común es la detección de un objeto específico en una imagen, para ello FLANN recibe como entrada dos vectores que contienen los descriptores obtenidos a partir de dos imágenes distintas, internamente se realiza el cálculo de las distancias entre dichos descriptores y se produce como salida un vector que contiene el resultado de este emparejamiento. El vector resultante contiene un conjunto de

registros donde cada uno se compone del identificador del descriptor base, el identificador del descriptor a emparejar y la distancia entre los descriptores. En base al resultado es posible filtrar los emparejamientos acertados para utilizarlos mediante técnicas de homografía (correspondencia entre puntos y rectas de dos figuras geométricas planas) y transformación de perspectiva con el fin de graficar los puntos clave y contornos de los objetos encontrados.

2.3. Modelamiento de material multimedia mediante tecnologías semánticas

Este proyecto de tesis contempla el procesamiento y análisis de video para luego obtener información acerca del contenido del mismo y almacenarla de manera que pueda ser posteriormente consultada. Tanto imágenes como video, se pueden catalogar como material multimedia. Para almacenar adecuadamente la información obtenida de su análisis, es necesario modelar formalmente dicho material, es decir, construir un modelo abstracto que represente eficazmente el concepto del mundo real y que a la vez sea legible por un computador. En este caso, se recurrirán a las tecnologías semánticas para realizar el proceso de modelado de armas de fuego así como el de anotación de videos.

Según [27] las tecnologías semánticas son herramientas que permiten obtener conocimiento a partir de las distintas asociaciones existentes en un conjunto de información, por lo que debe existir previamente un modelo de conocimiento dentro de un dominio específico al cual se le apliquen dichas herramientas en tiempo de ejecución.

A continuación se describirán algunas de las tecnologías semánticas que se usarán en la presente tesis.

2.3.1. Ontologías

Una ontología en términos generales se refiere a la naturaleza de la existencia de algo y a sus propiedades. Se puede definir, según [28], como una especificación de una conceptualización y desde este contexto, una ontología es una descripción de los conceptos y relaciones que puedan existir entre un agente o una comunidad de estos.

El concepto mencionado explica que una ontología es la representación formal de conceptos, propiedades y relaciones de situaciones, acciones u objetos existen-

tes en un dominio, es decir que una ontología modela situaciones de la vida real con el fin de crear una base de conocimiento.

Una ontología se encuentra formada por conceptos denominados “tripletas”; los cuales no son más que relaciones entre un sujeto, un predicado y un objeto. Las tripletas representan situaciones o acciones del mundo real, por ejemplo, el enunciado “Una persona tiene dinero” se puede representar en una tripleta de la siguiente manera:

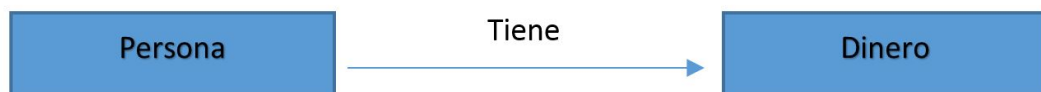


Figura 2.9: Representación de una tripleta

Donde el sujeto es “Persona”, el predicado es “Tiene” y el objeto es “Dinero”.

Según [29], las ontologías de acuerdo a su campo de aplicación se clasifican de la siguiente manera:

- **Ontologías de Alto Nivel.** Este tipo de ontologías describen conceptos bastante generales como es el caso del tiempo, materia, eventos, etc. Estos modelos son independientes del dominio en el cual se aplican ya que pueden ser usados en cualquier ámbito.
- **Ontologías de Dominio.** Estas ontologías poseen un vocabulario orientado a un dominio en específico como por ejemplo automóviles, armas de fuego, etc. Entiéndase por vocabulario a los nombres utilizados para definir genéricamente a sujetos, predicados y objetos dentro de una ontología.
- **Ontologías de Tarea.** Al igual que las ontologías de dominio este tipo de ontologías tienen su propio vocabulario pero orientado a tareas como por ejemplo el diagnóstico, las ventas, etc. Estos vocabularios se obtienen muy a menudo de la especialización de los términos de los vocabularios de las ontologías de alto nivel.
- **Ontologías de Aplicación.** Estas ontologías llegan a ser un modelo híbrido de conceptos obtenidos a partir de las ontologías de dominio así como de las de tarea, en donde se forma un modelo en el cual se especifican funciones desempeñadas por entidades de algún dominio específico en el desarrollo de alguna actividad.

En el presente proyecto de tesis se utilizarán ontologías de domino, en donde el primer paso consistirá en generar una ontología de armas de fuego para luego ser enlazada con una ontología para anotación de programas de televisión, todo esto con la finalidad de realizar el proceso completo de anotación semántica de armas de fuego existentes en videos.

2.3.2. Lenguajes para la representación de ontologías

Es necesario que las ontologías se encuentren representadas en lenguajes formales que puedan ser interpretados tanto por personas así como por máquinas. Estos lenguajes permiten dar significado e interpretar las relaciones que tienen las ontologías con la finalidad de obtener conocimiento a partir de estas.

Existen varios lenguajes formales para la representación de las ontologías. Dentro de los más importantes se encuentran los siguientes:

- **Resource Description Framework (RDF).** Es un estándar que fue creado por la *World Wide Web Consortium (W3C)*², el cual es utilizado para la descripción general y modelado de información de las ontologías. Es bastante parecido al *eXtensible Markup Language (XML)*, con la diferencia de que RDF está diseñado para conocimiento y no para datos, tal y como se explica en [30]. Esto quiere decir que RDF representa significado debido a que todo lo que se encuentra dentro de este son hechos comprobables basados en tripletas. Para más información relacionada a RDF se puede consultar [31] y [32].
- **Ontology Web Language (OWL)** Es un lenguaje formal que es utilizado para la descripción de las ontologías. Engloba a RDF pero es un lenguaje más extenso ya que da la facilidad para que se pueda interrelacionar ontologías con otras, inclusive con recursos no ontológicos (diccionarios, tesauros³, etc.). Más información sobre OWL se puede encontrar en [32] y [33].

Estos lenguajes serán utilizados para la representación formal de las ontologías en los Capítulos IV y V.

²<http://w3c.es/>

³Tesauros: Diccionarios de sinónimos

2.3.3. Repositorios semánticos

Un repositorio semántico según [34] es una herramienta que al igual que un *Sistema Gestor de Base de Datos* (SGBD) permite almacenamiento, consulta y manejo de datos estructurados; con la diferencia de que un repositorio semántico usa ontologías como esquema de datos con el fin de obtener un razonamiento automático a partir de los datos de consulta. Este tipo de repositorios está ideado específicamente para trabajar con datos que se encuentran dentro de un modelo ontológico haciendo que las consultas que se realizan sean rápidas y devuelva resultados de manera eficiente.

Se utilizará un repositorio semántico en el Capítulo V para almacenar el modelo ontológico de anotación de armas de fuego en video así como sus respectivas instancias, esto con la finalidad de poder realizar búsquedas.

2.3.4. SPARQL

SPARQL Protocol and RDF Query Language (SPARQL) es un lenguaje utilizado para realizar consultas sobre un modelo ontológico. Tal y como se explica en [35] el lenguaje de consulta SPARQL es utilizado para interrogar mediante consultas a varias fuentes de información, las mismas que pueden estar almacenadas en repositorios semánticos.

Este lenguaje de consulta además permite extender su ámbito de consultas debido a que tiene la capacidad de operar en varios grafos únicamente añadiendo prefijos (rutas) de otros dominios. Por otro lado, al igual que los lenguajes de bases de datos, SPARQL posee operadores de conjunción y disyunción para generar consultas más robustas y que devuelvan resultados más concretos. Los resultados que devuelve una consulta SPARQL son conjuntos de información generalmente ordenados en tripletas.

En el Capítulo V se hará uso de SPARQL para realizar consultas al repositorio semántico, con el fin de validar los resultados obtenidos durante el desarrollo de todo el proyecto de tesis.

2.4. Anotación semántica

Como se mencionó en las Secciones 2.3.1 y 2.3.2, las ontologías permiten modelar formalmente conceptos, vocabularios, etc. de manera que se pueda dar



un significado entendible por el computador. Lo cual, como resultado, facilitará el almacenamiento, extracción e interpretación de información.

2.4.1. Semántica ontológica

Antes de presentar los conceptos de anotación semántica, es necesario introducir lo que se conoce como Semántica Ontológica, la cual es una teoría dedicada al estudio del lenguaje natural, así como del *Procesamiento del Lenguaje Natural* (PLN), el cual representa al mundo real a través de un modelo abstracto. Luego utiliza el modelo como base para extraer y representar el significado del lenguaje, con el objetivo de realizar inferencia y razonar acerca del conocimiento que se pueda obtener [11].

2.4.2. Concepto y utilidad

Según la RAE, anotación es “la acción y efecto de anotar”, comúnmente utilizado para referirse al acto de colocar notas en un texto. Dentro del campo de las tecnologías semánticas, se puede considerar una anotación semántica como una información sobre las entidades o conceptos de una ontología [11]. Otro concepto se refiere a anotación semántica como un metadato acerca de contenido en general, ya sea texto, video, imágenes, etc., con el objetivo de ser enlazado a entidades o conceptos dentro de una ontología [36].

La función principal de una anotación semántica es relacionar contenidos con los conceptos de una ontología o brindar un punto de partida para que un computador comprenda el significado de un texto. El objetivo de anotar semánticamente los contenidos es facilitar los procesos de búsqueda y navegación, permitir el acceso inteligente a diferentes recursos según las necesidades, así como posibilitar el desarrollo de sistemas de inferencia que permitan interpretar la información adecuadamente [37].

2.4.3. Herramientas

Es posible anotar semánticamente, desde páginas web hasta cualquier tipo de texto o contenido. Existen diferentes herramientas para anotación semántica como COHSE⁴, Annotea⁵, Annozilla⁶, entre otras. Cada herramienta se aplica

⁴<http://users.ecs.soton.ac.uk/tmb/cohse/annotator/>

⁵<http://www.w3.org/2001/Annotea/>

⁶<http://annozilla.mozdev.org/>

para fines determinados y con diferentes resultados. Una de las maneras más comunes de realizar anotación semántica es almacenar anotaciones ya sea como objetos o propiedades correspondientes a un concepto dentro de una ontología, por lo que los editores de ontologías constituyen también herramientas poderosas para la anotación semántica [11]. Además, es posible realizar anotaciones para cierto fin, desarrollando una herramienta de anotación propia que cumpla las necesidades para las cuales ha sido pensada.

En este proyecto de tesis se desarrollará un prototipo que permita realizar anotaciones sobre videos, partiendo de los objetos encontrados en el módulo de reconocimiento. Dichas anotaciones serán modeladas mediante una ontología, con una o varias propiedades que permitan representar la relación existente entre los videos y los objetos encontrados en ellos. Finalmente las anotaciones serán almacenadas en un repositorio semántico para su posterior consulta.

2.5. Metodologías para la creación de ontologías

Es importante conocer que las ontologías no se crean de una manera empírica y que tampoco existe una sola forma de crearlas. Para este fin existen metodologías, entiéndase por metodología a una serie de pasos, procedimientos o fases que se deben seguir para cumplir un objetivo. Dichas metodologías han sido creadas a lo largo del tiempo, en donde cada una fue ideada para cubrir algún área de aplicación específica de diferentes temas como la medicina, la inteligencia artificial, sistemas gerenciales, entre otros [36]. Posteriormente han sido utilizadas como estándares para un dominio o grupos de dominio específicos.

Como se puede verificar en el trabajo realizado en [38], las principales metodologías para la creación de ontologías son:

- Metodología C Y C
- Metodología USCHOLD Y KING
- Metodología GRÜNINGER Y FOX
- Metodología KACTUS
- Metodología KACTUS
- Metodología NeOn

Estas metodologías han sido creadas para suplir múltiples áreas de interés y que pueden ser aplicadas de una manera general para la creación de ontologías. De igual manera en [38] se presentan ocho parámetros de evaluación para la selección de una metodología adecuada para la creación de ontologías, los mismos que son:

- Facilidad de Uso (FU).
- Polifuncionalidad por escenarios (PPE).
- Documentación (DOC).
- Reutilización (REU).
- Separación en los pasos (SP).
- Guías Metodológicas Claras (GM).
- Optimización de sus pasos (OP).
- Documentación Actualizada (DA).

Estos parámetros tienen un rango numérico de calificación que varía entre uno y tres; donde 1 representa BAJO, 2 representa MEDIO y 3 representa ALTO nivel de cumplimiento. Establecidos los parámetros y los respectivos rangos de calificación se procederá a crear una tabla de pesos en la cual se colocarán las metodologías con la respectiva calificación para cada parámetro. Al final se realizará una suma algebraica de los pesos obtenidos en cada parámetro para cada una de las metodologías, con el fin de obtener los puntajes totales para en base a ello seleccionar la mas adecuada.

En la Tabla 2.1 se pueden observar los resultados de dicha evaluación.

METODOLOGIA	FU	PPE	DOC	REU	SP	GM	OP	DA	TOTAL
CYC	2	1	2	3	2	1	3	1	15
USCHOLD Y KING	2	3	3	1	3	1	3	1	17
GRÜNINGER Y FOX	3	1	2	1	2	2	2	2	15
KACTUS	2	2	2	2	2	2	3	2	17
METHONTOLOGY	3	2	2	2	3	2	2	3	19
NeOn	3	3	3	3	3	3	3	3	24

Tabla 2.1: Análisis Comparativo de las Metodologías.

De los resultados obtenidos en la tabla anterior se observa que la metodología NeOn es la que obtuvo mayor puntaje por lo que se elegirá esta para el desarrollo de la ontología de las armas de fuego en el Capítulo IV.

Capítulo 3

Reconocimiento de Objetos en Video

El Reconocimiento de Objetos es un campo amplio, el mismo que hoy en día está siendo desarrollado e implementado en distintos ámbitos. Por ejemplo, la empresa CloudSight¹ ha desarrollado una aplicación móvil llamada CamFind² la misma que toma una foto con la cámara del dispositivo móvil y realiza una búsqueda en la Web devolviendo información relacionada al contenido de la fotografía. Dicha aplicación actualmente ha alcanzado más de 100 000 descargas, lo cual permite constatar la importancia que las personas le dan a este tipo de tecnologías. Además existen herramientas con un propósito más general para el reconocimiento de objetos como es el caso de FindObject³, la cual posee una gran cantidad de opciones y algoritmos para dicha tarea. Desafortunadamente estas aplicaciones no permiten realizar un reconocimiento en video de manera adecuada.

Con el objetivo de cumplir la primera fase del presente proyecto de tesis, en este capítulo se plantea el desarrollo de un módulo que permita reconocer armas de fuego en un video dado y registrar los intervalos de ocurrencia de las mismas en un archivo de datos que posteriormente se utilizará como entrada para el módulo de anotación semántica en el Capítulo V.

3.1. Selección del algoritmo para el proceso de reconocimiento de objetos

Como se trató en el Capítulo II, existe una gran variedad de algoritmos que pueden ser utilizados para la implementación de un módulo de reconocimiento.

¹<http://cloudsightapi.com/>

²<http://camfindapp.com/>

³<https://code.google.com/p/find-object/>

Para el prototipo que se pretende implementar, es necesario que el algoritmo que se elija permita desarrollar un módulo funcional que reconozca distintas armas de fuego que puedan aparecer en un video dado.

Luego de realizar la investigación para el contexto teórico del Capítulo II, se determinaron ciertos aspectos importantes que un algoritmo de reconocimiento debe cumplir:

- **Invariable a rotación.** El algoritmo debe permitir reconocer un objeto aun cuando este gire o se presente con un ángulo distinto al que se tiene como base.
- **Invariable a escala.** Es necesario poder reconocer un objeto a pesar de que varíe en tamaño. Esta situación se presenta generalmente cuando un objeto se acerca o aleja de la cámara en un video.
- **Invariable a inclinación.** Es pertinente que el algoritmo contemple variaciones de ángulo que presente el objeto debido al movimiento.
- **Invariable a obstrucción.** Permite reconocer un objeto a pesar de que pueda estar parcialmente cubierto.
- **Reconocimiento múltiple.** El algoritmo debe ser capaz de determinar la presencia de varios objetos a la vez, pues en caso de presentarse esta situación, es necesario reconocer todos los objetos presentes en una escena.
- **Inmune a cambios de luz.** Se necesita realizar reconocimiento a pesar de que se presenten variaciones en la luz.
- **Eficiencia.** El algoritmo debe ser eficiente en cuanto a los recursos que necesita para ejecutarse.

A cada uno de los parámetros mencionados se le asignará una calificación entre 1 y 5, siendo 5 el mayor puntaje, indicando que cumple con lo planteado en su totalidad y 1 el menor puntaje, indicando que cumple mínimamente con lo planteado. Las calificaciones han sido asignadas en base al criterio del equipo de trabajo.

Luego de analizar los algoritmos planteados en el Capítulo II, se observa en la Tabla 3.1 que el algoritmo SURF ha sido seleccionado como base para el desarrollo del módulo de reconocimiento de armas, pues cumple en mayor medida con los parámetros impuestos. Es importante recalcar que en este proyecto de tesis no

se plantea ninguna contribución a la mejora del algoritmo y por tanto al estado del arte en reconocimiento de objetos. El aporte de este proyecto consiste en juntar distintas herramientas y métodos para ejecutar el proceso de detección y reconocimiento de armas de fuego, así como la anotación semántica de videos.

	AFA	Meanshift	Camshift	SIFT	SURF
Invariable a rotación	3	1	1	5	5
Invariable a escala	4	1	5	5	5
Invariable a inclinación	2	1	1	1	2
Invariable a obstrucción	2	1	1	1	2
Reconocimiento múltiple	4	4	4	4	4
Resistente a cambios de luz	4	3	3	3	3
Eficiencia	2	4	4	3	4
TOTAL	21	15	19	22	25

Tabla 3.1: Análisis Comparativo de los algoritmos para reconocimiento de video.

El algoritmo FLANN al ser utilizado para tareas de emparejamiento de puntos y no para su detección queda fuera de la calificación presentada en la tabla. Sin embargo, SURF y FLANN comúnmente se combinan para tareas de reconocimiento. En este proyecto de tesis se aplicarán conjuntamente en el módulo de reconocimiento de armas de fuego en video, de manera que SURF permitirá la detección de puntos y descriptores, mientras que FLANN emparejará dichos puntos para determinar la presencia de un objeto.

3.2. Selección de la herramienta para el desarrollo del módulo de reconocimiento de objetos

En el Capítulo II se indicó que la visión computacional es una rama muy amplia que engloba diferentes áreas de estudio, en donde cada uno puede ser motivo de una especialización. Para estos fines existen herramientas de software encargadas de ayudar a los investigadores e interesados en el estudio de su área de aprendizaje o experticia.

Existen herramientas de propósito general, las mismas que pueden ser moldeables al gusto de la persona interesada; es decir, este tipo de herramientas generalmente tienen su código abierto permitiendo que se pueda usar para cualquier fin o ámbito. Por otro lado existen herramientas especializadas, orientadas

únicamente a un conjunto de áreas en específico como por ejemplo la medicina, la arquitectura, etc.

Para el desarrollo de la presente tesis se ha realizado un análisis de diferentes herramientas orientadas específicamente a trabajar con imágenes y videos y que además trabajen con algoritmos para reconocimiento de objetos. Para esto se han tomado en cuenta características generales que poseen las herramientas de software. El resultado se puede observar en la Tabla 3.2, donde se ha colocado cada una de las herramientas con la siguiente información:

- **Descripción.** Una breve reseña de la herramienta.
- **Características.** Tareas que se pueden realizar con la herramienta.
- **Ventajas.** Principales ventajas.
- **Desventajas.** Principales desventajas
- **Licencia.** El tipo de licenciamiento de la herramienta.
- **Portabilidad.** Cuales son los sistemas operativos en los cuales funciona la herramienta.
- **Interfaces de Programación.** Los lenguajes de programación en los cuales se puede implementar la herramienta.
- **Fuente.** Sitio web en el cual se puede encontrar más información y documentación sobre la herramienta.

En la Tabla 3.2 se puede observar cuatro herramientas, las cuales tienen ventajas y desventajas para el manejo de computación visual y reconocimiento de objetos.

Para la selección de la herramienta óptima para el desarrollo del módulo de reconocimiento de objetos en un video se han planteado los siguientes parámetros:

- Licencia (L)
- Portabilidad (P)
- Lenguajes de Programación (LP)
- Documentación (D)
- Algoritmos de Reconocimiento (AR)

Nombre	Descripción	Características	Ventajas	Desventajas	Licencia	Portabilidad	Lenguajes de Programación	Fuente
OpenCV	OpenCV es un conjunto de librerías orientadas a la visión computacional y machine learning. Fue construido con el fin de proveer una infraestructura común para las aplicaciones de visión computacional.	*Esta desarrollado en C++ y fue creado por Intel. *Permite realizar reconocimiento de imágenes, seguimiento de objetos etc. *Se puede generar aplicaciones móviles para Android. *Trabaja con una gran cantidad de lenguajes de programación, permitiendo acoplarse a las necesidades de los programadores.	*Tiene una gran cantidad de algoritmos optimizados que cubren todos los campos de la visión computacional. *Ampliamente usado por empresas a nivel mundial tales como Google, Toyota, Sony etc.	No es un producto final, es una herramienta para desarrollo.	BSD	GNU/Linux, Mac OS X y Windows	C++, C, Python, Java y MATLAB	http://opencv.org/
Computer Vision System Toolbox	Herramienta propietaria de MATLAB. Provee algoritmos, funciones y aplicaciones para procesamiento de videos y visión computacional.	Detección y seguimiento de objetos. Calibración de cámara. Herramientas de procesamiento de video e imágenes. Algoritmos disponibles como funciones de MATLAB, objetos System y bloques de Simulink. Herramientas para anotación de objetos.	Herramientas para anotación de objetos y trazado de gráficos. Cuenta con varios algoritmos pre entrenados para detección y seguimiento de objetos.	Software propietario. Requiere MATLAB o Image Processing Toolbox para su funcionamiento. No es un producto final, es una herramienta para desarrollo.	Personal: \$1700 mas impuestos Grupos: No disponibles en línea Estudiantes: \$45 por Matlab mas adicional por c/complemento	GNU/Linux, Mac OS X y Windows	MATLAB Soporta generación de código C automático.	http://www.mathworks.com/products/computer-vision/
Fast CV	FastCV es una librería para visión computacional desarrollada por Qualcomm, la misma que esta orientada a programadores que deseen realizar aplicaciones móviles de visión computacional.	*Posee un propio SDK llamado Snapdragon.	* Sus librerías están optimizadas para ocupar pocos recursos con procesadores Qualcomm, haciendo que las apps sean rápidas en estas arquitecturas	Únicamente funciona para generar aplicaciones móviles. *Las aplicaciones generadas son solo para Android.	GPL	GNU/Linux, Mac OS X y Windows	Java	https://developer.qualcomm.com/mobile-development/advanced-features/computer-vision-fastcv
Robo Realm	Es una aplicación usada para visión computacional, análisis de imágenes y video. Usado ampliamente en sistemas robóticos.	Modalidad de trabajo desde GUI o API. Algoritmos para reconocimiento de objetos y etiquetado. Algoritmos para procesamiento de imagen y video. Ambiente de simulación para pruebas. Entradas y salidas de distinto tipo.	Variedad de filtros y algoritmos pre compilados para uso directo. Interfaz amigable. API ampliamente soportada por servidores. Varias fuentes de imágenes y videos. Múltiples interfaces de salida: file, web, ftp, etc. Posibilidad de personalización a través de plugins.	Dificultad de integración con código externo. Documentación limitada. Necesidad de desarrollo de plugins propios para funcionar como interfaz con otros lenguajes.	Personal \$49.95 Académica \$39.95 Comercial \$159.95	Windows	Scripting con VBScript, Python y CScript.	http://www.roborealm.com/help/index.php

Tabla 3.2: Herramientas para el análisis de imágenes y video

Con el fin de seleccionar la herramienta adecuada se calificará a cada parámetro con valores que oscilan entre el 1 y 3 representando lo siguiente:

- **Valor de “1”**. Si su licenciamiento es pagado, no es portable, permite el desarrollo en un lenguaje de programación, no tiene documentación oficial y manejan un algoritmo específico de reconocimiento.
- **Valor de “2”**. Si su licenciamiento es libre pero tiene versiones pagadas, su portabilidad cubre varios sistemas operativos (sin incluir sistemas operativos para móviles), permite el desarrollo en hasta tres lenguajes de programación, su documentación es básica (no incluye ejemplos de aplicación) y manejan hasta tres algoritmos de reconocimiento
- **Valor de “3”**. Si su licenciamiento es libre, su portabilidad cubre varios sistemas operativos (incluyendo sistemas operativos para móviles), permite el desarrollo en más de tres lenguajes de programación, su documentación es bastante extensa (incluye ejemplos de aplicación) y manejan muchos algoritmos de reconocimiento.

Los resultados de calificación de cada parámetro para cada herramienta se pueden observar en la Tabla 3.3.

Herramienta	L	P	LP	D	A	TOTAL
OpenCV	3	3	3	3	3	15
Computer Vision System Toolbox	1	3	1	3	3	11
Fast CV	3	3	1	2	2	11
RoboRealm	2	1	2	3	3	11

Tabla 3.3: Tabla comparativa de herramientas para la visión computacional.

De los resultados se observa que OpenCV tiene el mayor puntaje de pesos en la tabla, lo cual indica que esta herramienta es la ideal para el desarrollo del presente proyecto de tesis.

3.3. Desarrollo de un prototipo para el reconocimiento de armas en video

En esta sección se describirá el proceso realizado para el desarrollo del módulo de reconocimiento de objetos en video. Teniendo en cuenta que este proyecto

podría ser tomado como base para trabajos futuros o de índole semejante, se describirá el proceso de manera general para servir como guía al lector.

El módulo tiene por objetivo recibir como entrada un video en el cual se muestren armas de fuego, mediante los algoritmos elegidos se realizará el reconocimiento de las armas y se producirá como salida un archivo que contendrá la información de los objetos detectados.

En primer lugar, cabe recordar que como se mencionó en la Sección 1.2, esta tesis es parte del proyecto “Empleo de tecnologías semánticas para el análisis de contenido multimedia transmitido para televisión digital terrestre”. En el proyecto, una de las actividades plantea el proceso de captación de la señal y por ende de los videos que se trasmiten en la televisión digital. Por ello, se asumirá que se cuenta con un directorio de videos en formato .mp4, así como una base de datos donde se puede encontrar la información básica correspondiente a cada video. Dichos videos serán utilizados como entrada del módulo de reconocimiento.

En la Figura 3.1 se muestra un esquema representativo del módulo de reconocimiento en donde se observa un video como entrada, luego se aplicarán principalmente los algoritmos SURF y FLANN para realizar el reconocimiento de acuerdo a un dataset de imágenes modelo, finalmente se produce una salida en formato .xml con los resultados obtenidos. Se pretende entonces desarrollar un módulo para el reconocimiento de armas de fuego y realizar un conjunto de pruebas controladas para comprobar su funcionalidad.

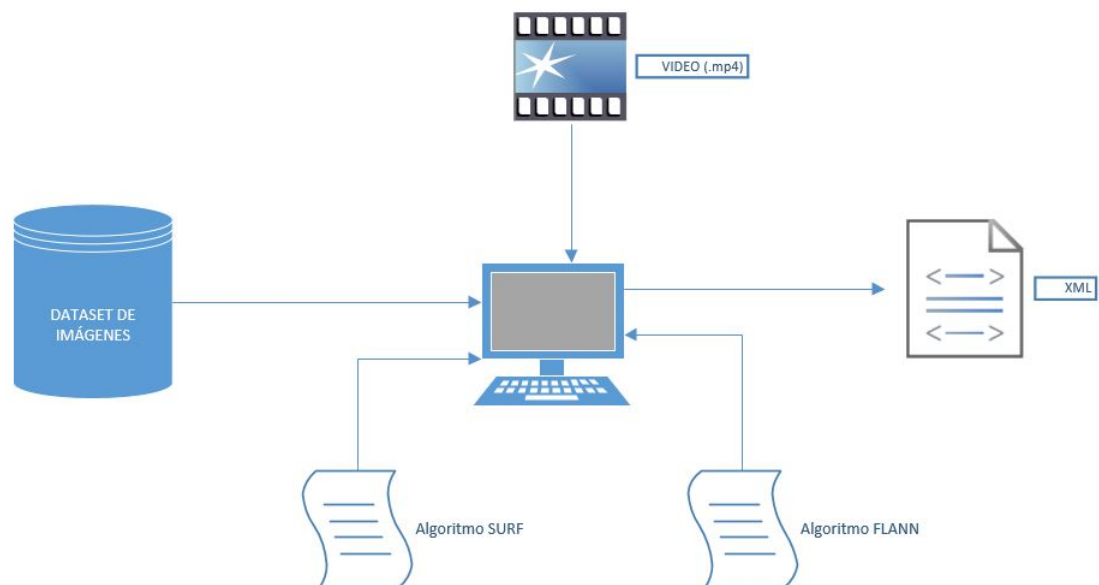


Figura 3.1: Esquema del módulo de reconocimiento.

3.3.1. Entradas y salidas

A continuación se describe brevemente las características de los archivos que se utilizaron como entradas y salidas del módulo de reconocimiento de objetos.

- **Video de entrada.** Los videos utilizados como entrada tienen formato .mp4. Sus dimensiones son 1920 x 1080 píxeles. pero se podrán utilizar también videos de mayores o menores dimensiones siempre y cuando no presenten distorsiones en la imagen. Para efectos de este prototipo, los videos deben estar filmados bajo buenas condiciones de iluminación, preferiblemente bajo luz natural durante el día y no presentar gran variación de luz, así como evitar que las armas se filmen a contraluz pues esto disminuiría el contraste entre el arma y el entorno que la rodea causando errores en el resultado del reconocimiento.
- **Dataset de imágenes.** Las imágenes a utilizarse como modelos base para el reconocimiento de armas pueden ser a colores o en escala de grises, deben enfocar correctamente al objeto a reconocer, ser recortadas para contener solamente el objeto, con un ancho mínimo de 150 px y un largo variante dependiendo de la geometría del objeto, no deben estar borrosas o pixeladas, y deben utilizar un fondo neutro, es decir no contener otros elementos o patrones que puedan ser considerados erróneamente como parte el objeto. Para efectos del desarrollo de esta tesis se ha utilizado un dataset que contiene tres imágenes las cuales han sido cargadas en una base de datos MySQL conteniendo la *Uniform Resource Locator* (URL) de la imagen, así como datos básicos para identificarlas.
- **Archivo XML.** Una vez realizado el proceso de reconocimiento, como salida los resultados deben ser almacenados para su posterior uso. Se decidió generar un archivo .xml debido a la facilidad de construcción y a la facilidad de lectura que proporciona su estructura. La estructura definida para el archivo se describe en el Listing 3.1.

Se tendrá una lista de objetos reconocidos delimitada con las etiquetas `< Objetos >< /Objetos >`. Cada objeto reconocido se definirá dentro de las etiquetas `< Objeto >< /Objeto >`. Cada objeto tendrá un identificador entre las etiquetas `< Identificador >< /Identificador >` el identificador del video en el que aparece entre las etiquetas `< IdVideo >< /IdVideo >`, el nombre del objeto entre las etiquetas `< Nombre >< /Nombre >`, el

momento en el que aparece el objeto en milisegundos entre las etiquetas `< Inicio >` `< /Inicio >` y por último, el momento en el que desaparece del video en milisegundos entre las etiquetas `< Fin >` `< /Fin >`. Cabe mencionar que los identificadores tanto de los objetos como de los videos, son los identificadores secuenciales únicos asignados a los videos y objetos en la base de datos modelo utilizada como entrada para el módulo.

```
<Objetos>
  <Objeto>
    <Identificador>4</Identificador>
    <IdVideo>3</IdVideo>
    <Nombre>Automag</Nombre>
    <Inicio>5520</Inicio>
    <Fin>5680</Fin>
  </Objeto>
</Objetos>
```

Segmento de código 3.1: Estructura del archivo .xml

3.3.2. Herramientas de desarrollo

Para realizar la tarea de programación del prototipo de reconocimiento explicado anteriormente se utilizaron las herramientas que se describen brevemente a continuación.

QT Creator⁴

Este software es un *Integrated Development Environment* (IDE) de desarrollo, el mismo que permite codificar aplicaciones en varios lenguajes de programación. También permite la generación de interfaces graficas así como la facilidad para ejecutar aplicaciones mediante consola únicamente. Está disponible para usarse en GNU/Linux⁵, Mac OS X⁶ y Windows⁷.

OpenCV⁸

Es una herramienta de software que proporciona un conjunto de librerías, las cuales poseen un gran número de algoritmos que permiten realizar tareas de re-

⁴<https://www.qt.io/>

⁵<http://www.gnu.org/gnu/linuxandgnu.html>

⁶<https://www.apple.com/osx/>

⁷<https://www.microsoft.com/enus/windows>

⁸<http://opencv.org/>

conocimiento y procesamiento tanto en imágenes así como en videos. Posee un licenciamiento *Berkeley Software Distribution* (BSD)⁹, se puede integrar con muchos IDEs de desarrollo y está disponible para su uso en GNU/Linux, Mac OS X y Windows.

MySQL¹⁰

Es un SGBD que permite el almacenamiento, mantenimiento y consulta de información. Es uno de los SGBD más populares debido a su facilidad de uso y a las múltiples herramientas que proporciona para el manejo de información. Funciona sobre GNU/Linux, Mac OS X y Windows.

C++

Es un lenguaje de programación basado en C. Este lenguaje a diferencia de C permite la manipulación de objetos convirtiéndolo así en un lenguaje de programación multiparadigma debido a que también maneja programación estructurada.

3.3.3. Desarrollo

Una vez definidas las entradas y las salidas del módulo, conociendo los algoritmos a utilizar y las herramientas que proporcionan dichos algoritmos, se desarrolló el módulo de reconocimiento de armas en video.

A continuación se describe brevemente el proceso realizado para desarrollar el módulo de reconocimiento:

1. Descarga e instalación de librerías de OpenCV en Qt Creator. Ver Anexo A.
2. Diseño de la interfaz incluyendo: Selección del video, carga del video, análisis y reconocimiento de objetos, almacenamiento de resultados.
3. Entrada y procesamiento de video: Consiste en leer un archivo de video del sistema de archivos del computador. Básicamente, se utilizó un temporizador para tomar cada cuadro o frame del video. Y finalmente se realizó una transformación de la imagen a escala de grises para evitar posibles errores a causa de la luz.

⁹<http://opensource.org/licenses/BSD3Clause>

¹⁰<https://www.mysql.com/>

4. Reconocimiento: El proceso consiste en tomar la imagen y detectar los puntos clave a través del algoritmo SURF, así también se detectarán los puntos clave de los objetos presentes en cada una de las imágenes del dataset. Luego se extraen los descriptores de cada punto, igualmente utilizando SURF.

Una vez hecho lo anterior, se procederá a emparejar los descriptores de la imagen con los descriptores de los objetos, de manera que se detecte si se presenta o no una ocurrencia de un objeto en la imagen. Esto se realiza mediante el algoritmo FLANN. Una vez detectados los descriptores que coinciden, se recuperan nuevamente los puntos clave de la imagen de entrada y se procede a graficar el contorno del objeto encontrado.

5. Almacenamiento de datos: A medida que se analizan las imágenes, se almacena las ocurrencias de cada objeto, para luego definir intervalos de aparición. Finalmente, se almacena en el archivo .xml antes mencionado cada uno de los objetos (armas) detectados con los respectivos datos de inicio y fin del intervalo de aparición, así como el identificador del video en el que aparece.

3.4. Resultados

Una vez desarrollado el módulo de reconocimiento, se realizaron varias pruebas para comprobar la funcionalidad del mismo. Ver proceso de ejecución en Anexo B.

Los resultados obtenidos del proceso de ejecución con varios videos de prueba y utilizando diferentes combinaciones de armas se encuentran anotados en la Tabla 3.4, donde se observa el nombre del video, su duración, intervalos reconocidos tomados del archivo de salida de cada ejecución, intervalos reales obtenidos directamente de los videos y, por último, tiempo de ejecución transcurrido durante el proceso de reconocimiento en cada video.

Posteriormente se analizaron los resultados obtenidos para calcular un porcentaje de efectividad que permita validar el funcionamiento adecuado del módulo de reconocimiento.

El porcentaje fue calculado mediante la siguiente formula:

$$\text{Porcentaje de Efectividad} = \frac{\text{Intervalo Reconocido} * 100}{\text{Intervalo Real}} \quad (3.1)$$

En donde:

- **Intervalo Reconocido.** Es el total en milisegundos de las apariciones reconocidas por el módulo.
- **Intervalo Real.** Es el total en milisegundos de las apariciones reales en el video. Estos intervalos fueron determinados por el equipo de trabajo mediante la observación minuciosa de los videos.

Para el caso de los videos en los que aparece más de un arma se realizó el promedio de los porcentajes de efectividad. Los resultados se pueden observar en la Tabla 3.5.

Del análisis anterior se determinó que el porcentaje promedio de exactitud alcanza el 90 %, lo cual es satisfactorio e indica que se ha cumplido exitosamente con el objetivo específico de realizar el reconocimiento de un conjunto determinado de armas de fuego en un video específico. Si bien los tiempos de ejecución aún son altos con respecto a la duración de los videos, la optimización de los algoritmos será planteada en la Sección 2.6 Futuras Líneas de Investigación con el propósito de producir una segunda versión del prototipo que devuelva mejores resultados.

Finalmente, de todo el proceso realizado, la información obtenida como salida en archivos .xml servirá como entrada para el prototipo de anotación de material multimedia tratado en el Capítulo V.

Video	Duración del video	Intervalos reconocidos	Intervalos reales	Tiempo de ejecución
	(hh:mm:ss)	(ms.)	(ms.)	(mm:ss,ff)
Video con revolver CO2 Dan Wesson 6"	0:00:26	3440 – 7800 12040 – 17080 21280 – 24800	3440 – 7900 12000 – 17100 21200 – 24850	07:45,7
Video con pistola Beretta 92 FS	0:00:26	3840 – 8800 13280 – 16640 19240 – 24800	3800 – 8800 13240 – 16720 19240 – 24800	07:51,8
Video con escopeta Mossberg 590 Mariner	0:00:29	2920 – 9560 14600 – 19800 23560 – 28400	2920 – 9600 14640 – 19850 23800 – 28450	07:55,8
Video con revolver CO2 Dan Wesson 6" con giro	0:00:20	991 – 19404	200 – 20000	07:59,1
Video con pistola Beretta 92 FS con giro	0:00:36	1120 – 9720 12520 – 16040 19200 – 24480 27320 – 35160	1180 – 15960 18500 – 35800	11:20,2
Video con escopeta Mossberg 590 Mariner con giro	0:00:36	2600 – 11000 16400 – 20600 25960 – 2900 31280 – 33320	2600 – 21800 25800 – 35500	11:04,6
Video con dos armas: pistola Beretta 92 FS, revolver CO2 Dan Wesson 6"	0:00:28	Revolver 8330 - 13586 16198 - 20793 22380 - 27206 Pistola 3537 - 9586 16528 - 24099	Revolver 8300 - 13590 6198 - 27300 Pistola 3500 - 9590 16500 - 24150	07:55,8
Video con tres armas: pistola Beretta 92 FS, revolver CO2 Dan Wesson 6", escopeta Mossberg 590 Mariner con giro	0:00:31	Revolver 3966 – 9454 21785 - 28958 Pistola 4892 – 9553 21553 - 28628 Escopeta 8793 -16264 21619 - 28264	Revolver 3950- 9500 21780- 28960 Pistola 4850 - 9600 21550 - 28640 Escopeta 8700 - 16300 21600- 28300	09:40,0

Tabla 3.4: Tabla de resultados del módulo de las pruebas realizadas.

Video	Porcentaje de Efectividad
Video con revolver CO2 Dan Wesson 6"	97.67 %
Video con pistola Beretta 92 FS	98.58 %
Video con escopeta Mossberg 590 Mariner	98.74 %
Video con revolver CO2 Dan Wesson 6" con giro	92.99 %
Video con pistola Beretta 92 FS con giro	78.45 %
Video con escopeta Mossberg 590 Mariner con giro	59 %
Video con dos armas: pistola Beretta 92 FS, revolver CO2 Dan Wesson 6"	95.63 %
Video con tres armas: pistola Beretta 92 FS, revolver CO2 Dan Wesson 6", escopeta Mossberg 590 Mariner con giro	99.03
TOTAL	90.01 %

Tabla 3.5: Porcentaje de efectividad de reconocimiento.

Capítulo 4

Representación Formal de Objetos

Las ontologías son modelos formales de datos. Actualmente, se utilizan ontologías para modelar situaciones del mundo real que no pueden ser representadas fácilmente mediante los métodos convencionales. La ventaja de usar este tipo de tecnologías semánticas radica en la capacidad de enriquecimiento continuo de la información, así como la facilidad de integración entre distintas fuentes. Son varias las instituciones y entidades de distinta índole que han decidido adoptar las tecnologías semánticas para el manejo de su información debido a su eficacia y eficiencia. Un ejemplo de esto es la red social Facebook¹ con su plataforma Open Graph² la cual permite hacer referencia a conexiones a través de perfiles de usuarios. Por otro lado la aplicación Flipboard³ utiliza tecnologías semánticas para implementar buscadores de noticias e intereses personalizados mediante tendencias y en tiempo real. Esto demuestra que las ontologías están siendo ampliamente utilizadas en la actualidad por empresas de vanguardia, proporcionando una visión acertada del potencial que presentan las tecnologías semánticas al ser implementadas en proyectos informáticos.

Este capítulo comprende el modelado formal de armas de fuego a través de una metodología que permita aprovechar los recursos existentes sobre esta temática. La finalidad es obtener una ontología que modele armas de fuego con una clasificación adecuada de las mismas. Dicha ontología contendrá varias instancias de armas de fuego; por otro lado el módulo de reconocimiento planteado en el Capítulo III reconocerá algunas de estas armas en videos que presentan escenas violentas, esto en conjunto servirá para realizar la anotación semántica de videos en el Capítulo V.

¹<https://www.facebook.com/>

²<http://ogp.me/>

³<https://flipboard.com/>

4.1. Recursos ontológicos y no ontológicos

Como paso inicial para el desarrollo de la ontología de armas de fuego se debe conocer cuáles son los recursos de los que se obtendrá información relacionada a esta temática. Dichos recursos pueden ofrecer datos importantes como características, clasificaciones, nombres, sinónimos, etc. Es importante indicar que existen dos tipos de recursos que pueden ser involucrados dentro de la creación de una ontología que son:

- **Recursos Ontológicos (OR).** Estos recursos están conformados por ontologías o conjuntos de ellas cuya semántica está formalmente descrita. Este tipo de recursos se encuentran en buscadores semánticos tales como Swoogle⁴, Watson⁵, entre otros. Se pueden comunicar y reusar con otras ontologías gracias a que poseen un lenguaje formal en común y sus términos se pueden interrelacionar con la finalidad de enriquecer los modelos ontológicos.
- **Recursos No Ontológicos (NOR).** Según [39] los recursos no ontológicos son aquellos conjuntos de conocimiento cuya semántica no se encuentra formalizada por una ontología. Estos se pueden presentar en forma de texto, esquemas de clasificación, léxicos, tesauros y demás. A pesar de ser recursos que no están formalmente modelados pueden ser utilizados para enriquecer el conocimiento de un modelo ontológico y permitir que las búsquedas ontológicas sean más eficaces.

En [7] se encuentra una clasificación y categorización de los NOR en base a su tipo, tal y como se muestra en la Figura 4.1. En dicha figura se pueden observar tres capas. La primera corresponde a los diferentes tipos de recursos ontológicos existentes. La segunda capa indica los diferentes modelos de datos en los que puede ser implementado un recurso en específico; se observa que para el Esquema de Clasificación (Classification Scheme) se tienen cuatro modelos de datos en los cuales representar la información. Por último, la tercera capa indica el formato de implementación (base de datos, XML, archivos planos, hojas de cálculo) que se utilizará para generar finalmente un recurso no ontológico que pueda ser reutilizado para la creación de una ontología.

⁴<http://swoogle.umbc.edu/>

⁵<http://watson.kmi.open.ac.uk/WatsonWUI/>

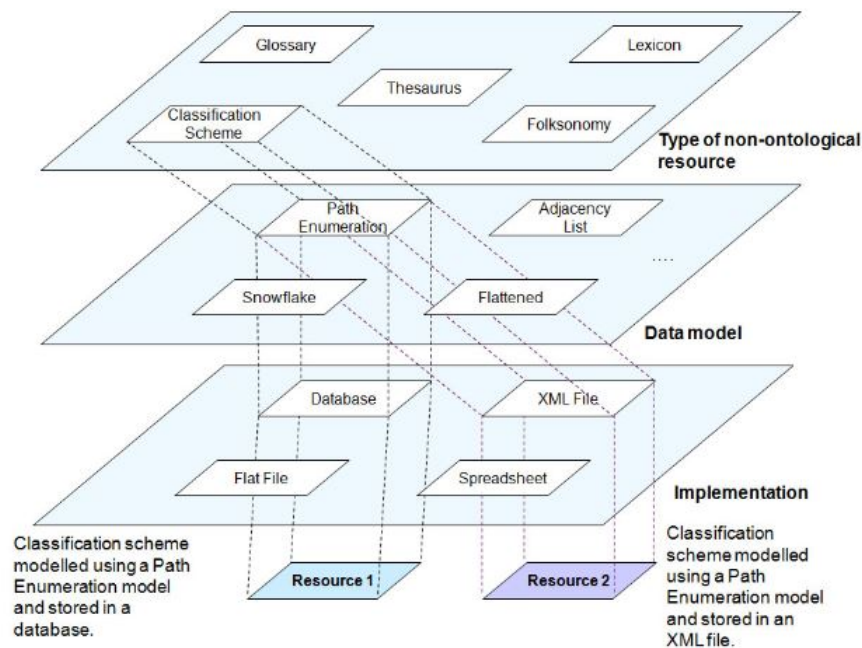


Figura 4.1: Categorización de los Recursos No Ontológicos [7].

En el presente proyecto de tesis se utilizarán recursos ontológicos para la creación de una ontología de armas de fuego basado en una metodología que plantea la reutilización de dichos recursos.

4.2. Especificación de requerimientos de la ontología

El siguiente paso consiste en la creación del modelo semántico ya sea en base a la reutilización de ontologías existentes o a la creación de nuevas ontologías. Dichas ontologías deben permitir el modelamiento de una clasificación de armas de fuego, el punto de partida para ello será la información extraída de sitios altamente confiables en Internet así como la ontología referente a armas planteada en [16].

Del Capítulo II en la Sección 2.5 se obtuvo que la metodología NeOn es la más adecuada para la creación de la ontología planteada anteriormente, la cual permitirá cubrir la actividad inicial de especificación de requisitos de la ontología, para luego proseguir con el modelamiento de la misma y finalmente su aplicación en el contexto planteado.

Utilizando la plantilla del documento de especificación de requisitos propues-

ta por la metodología NeOn, se procederá a describir cada una de las tareas realizadas con la finalidad de generar una especificación de requerimientos.

1	Propósito
El propósito de la construcción de la ontología de armas de fuego es proveer un modelo general de clasificación de las mismas, con el fin de que posteriormente pueda ser utilizado para la anotación de videos en los cuales aparezcan dichas armas.	
2	Alcance
La ontología tiene como alcance modelar una clasificación ampliamente aceptada de los distintos tipos de armas, tomando en cuenta características como peso, longitud, munición que utiliza, fabricante, etc.	
3	Lenguaje de Implementación
La ontología será implementada en OWL/glsrdf y almacenada en un repositorio semántico.	
4	Usuarios Finales Previstos
Usuario 1: Televidente que desee buscar videos que contengan armas de fuego. Usuario 2: Agencias Televisivas.	
5	Usos Previstos
Uso 1: Anotación de videos que contengan armas de fuego. Uso 2: Consulta de información sobre armas de fuego presentes en videos.	

Tabla 4.1: Documento de Especificación de Requisitos: Secciones 1 a 5.

En la Tabla 4.1 se observan las primeras cinco secciones del documento de especificación de requerimientos. El propósito, alcance y lenguaje de implementación fueron elegidos de acuerdo a las necesidades y conceptos referentes a los objetivos del presente proyecto de tesis por parte del equipo de trabajo. Por otro lado, tanto los usuarios finales previstos así como los usos previstos fueron obtenidos en base a la necesidad que pudiesen tener ciertos usuarios para reconocimiento de armas de fuego en videos en general.

De la Tabla 4.1 se obtuvo las temáticas indispensables para determinar los requerimientos que debe cumplir la ontología con el fin de que funcione adecuadamente; por lo tanto la sección 6 de esta plantilla cubre todo lo relacionado a requerimientos funcionales y no funcionales de la ontología. Los requerimientos no funcionales hacen referencia a los requerimientos que no pueden ser medidos de manera cuantitativa, mientras que los requerimientos funcionales son todos aquellos que pueden ser medidos cuantitativamente mediante algún proceso.

Para el caso específico de los requerimientos funcionales se usó el método de preguntas de competencia [40], el mismo que consiste en escribir preguntas la ontología a crear sea capaz de responder.

6	Requerimientos de la Ontología
a)	Requerimientos No Funcionales
	RNF1. La ontología debe soportar un escenario multilingüe en los idiomas: Español e Inglés. RNF2. La ontología debe estar basada en modelos o clasificaciones de armas ya conocidos.
b)	Requerimientos Funcionales – Preguntas de Competencia
	PC1. ¿Cuál es el nombre del arma? AK-47, Beretta ARX-160, CZ Duo, ... PC2. ¿Cuál es el lugar de origen del arma? USA, Londres, Rusia, ... PC3. ¿Cuál es el fabricante del arma? Izhmash, Beretta, Duo, ... PC4. ¿Cuál es el peso del arma? 4.3 kg, 0.4 kg, 3.1 kg PC5. ¿Cuál es la longitud del arma? 870 mm, 755 mm, 111 mm, ... PC6. ¿Cuál es el calibre del arma? 415 mm, 5.56mm, 6.35 mm, ... PC7. ¿Cuál es la categoría a la que pertenece el arma? Pistola, Rifle, Escopeta, ... PC8. ¿Cuál es el sistema de disparo del arma? Automatico, Semi-Automatico, Manual, etc

Tabla 4.2: Documento de Especificación de Requisitos: Sección 6.

Como se observa en la Tabla 4.2, se obtuvo dos requerimientos no funcionales basados en la experiencia que se tiene de trabajar con ontologías, ya que el objetivo es tener un escenario multilingüe y además que la ontología se creará en base a fuentes ya existentes con el fin de reutilizar esa información para enriquecer a la ontología a ser construida. En cuanto a los requerimientos funcionales, se realizaron preguntas a usuarios que comúnmente ven videos, películas y series de acción en las cuales aparecen armas con la finalidad de obtener información sobre todo lo que concierne a la ontología que se creará.

Hecho todo lo anterior se procede a generar la Sección 7, que corresponde a un pre-glosario de términos obtenido a partir de las respuestas dadas por las personas entrevistadas en la sección 6.b sobre las preguntas de competencia planteadas.

Se dividió a esta sección en tres subsecciones, en la sub-sección 7.a se colocaron todos los términos obtenidos desde las preguntas de competencia con el respectivo número de ocurrencias de las mismas. En la sub-sección 7.b se colocaron todos los términos que fueron obtenidos a partir de las respuestas a cada una de las preguntas de competencia y de la misma forma se indica su número de ocurrencias. Finalmente la sub-sección 7.c contiene objetos que han sido identificados en las preguntas de competencia, entiéndase por objetos a los nombres propios, lugares, empresas, marcas, etc.

Como resultado de este proceso se observó que los términos encontrados se

7	Pre-Glosario de Términos
a)	Términos desde las preguntas de competencia
Arma	9
Nombre	1
Lugar de Origen	1
Cañón	3
Fabricante	1
Peso	1
Longitud	2
Calibre	1
Sistema de Disparo	1
Categoría	1
b)	Términos desde las respuestas a las preguntas de competencia
USA, Londres, Rusia, ...	1
Izhmash, Beretta, Duo, ...	1
Automático, Semi – Automático, Manual...	1
Pistola, Rifle, Escopeta, ...	1
c)	Objetos
USA, Londres, Rusia, ...	
Izhmash, Beretta, Duo, ...	

Tabla 4.3: Documento de Especificación de Requisitos - Sección 7.

refieren a características propias de cada arma, así como su información de origen (fabricante y lugar de fabricación del arma). Esta información deberá ser modelada en la ontología tomando como referencia las fuentes externas que pudiesen ser reusadas para este fin.

Es necesario categorizar la terminología con miras a la formación de la estructura de la nueva ontología. Las categorías identificadas fueron las siguientes:

- **Información del arma.** Esta categoría contiene información propia del arma, es decir, sus características principales como: nombre, peso, calibre, longitud, etc.
- **Información de Origen.** En esta categoría se identifica la procedencia del arma tomando en cuenta tanto a su lugar de fabricación así como a su fabricante.
- **Categorías de Armas de Fuego.** En base a las preguntas de competencia se hizo evidente que existe una gran variedad de armas de fuego y que su clasificación no está definida de acuerdo a un estándar específico, por lo que posteriormente se buscará una clasificación en base a fuentes externas.

Fruto del análisis anterior, las categorías encontradas servirán para determinar los recursos que pueden ser reutilizados en la generación de la ontología de armas de fuego.

4.3. Modelado de la ontología de armas de fuego.

Al momento de crear una ontología es necesario seguir una metodología que permita obtener resultados eficientes de acuerdo al dominio específico en el cual se aplicará. Por lo tanto, como se indicó en la Sección 4.2 se usará la metodología NeOn, la misma que fue seleccionada debido a que alcanzó el puntaje más alto en el análisis realizado en el Capítulo II, Sección 2.5 .

La metodología NeOn propuesta en [7] es una metodología basada en escenarios y que básicamente está orientada a la reutilización y reuso tanto de recursos ontológicos así como no ontológicos para la generación de ontologías.

Dicha metodología comprende los siguientes escenarios:

- **Escenario 1.** Desde la especificación de la aplicación.
- **Escenario 2.** Reutilización y reingeniería de los recursos no ontológicos.
- **Escenario 3.** Reutilización de los recursos ontológicos.
- **Escenario 4.** Reutilización y re-ingeniería de los recursos ontológicos.
- **Escenario 5.** Reutilización y fusión de los recursos ontológicos.
- **Escenario 6.** Reutilización, fusión y re-ingeniería de los recursos ontológicos.
- **Escenario 7.** Reutilización de *Patrones de Diseño de Ontologías* (ODP).
- **Escenario 8.** Reestructuración de recursos ontológicos.
- **Escenario 9.** Localización de recursos ontológicos.

De este conjunto de escenarios se seleccionó el escenario número 6 ya que se reutilizarán recursos ontológicos y se los fusionará para obtener una ontología a la cual finalmente se le aplicará un proceso de reingeniería para cubrir todas las necesidades que se requieren en el modelo ontológico a crear.

Con la finalidad de cumplir con dicho escenario de la metodología NeOn se describen a continuación las actividades desarrolladas:

1. **Búsqueda de ontologías.** Como tarea inicial se realizó la búsqueda de ontologías que cumplan con todos los requisitos especificados en la Sección 4.2 en el Documento de Especificación de Requerimiento de la Ontología. Dicha actividad se realizó utilizando motores de búsqueda orientados a encontrar recursos ontológicos tales como Watson, Swoogle, Falcons⁶ y Sindice⁷.

Como resultado de esta búsqueda se obtuvo que las ontologías Weapon Ontology⁸, DBpedia Ontology⁹, IPTC News Codes¹⁰, NERD¹¹ y WordNet¹² contienen información relacionada a las armas de fuego. En la Tabla 4.4 se listan cada una de ellas así como el proyecto o institución creador de dicho modelo ontológico.

Ontología	Proyecto o Institución
Categoría 1: Ontologías relacionadas con armas de fuego	
Weapon Ontology	A. Arslan, N. Sirakov y S. Attardo
DBpedia Ontology	Comunidad Wikipedia y Otros
IPTC News Codes	International Press
NERD	G. Rizzo Y R. Troncy
WordNet	Ontologies

Tabla 4.4: Ontologías relacionadas con armas de fuego.

2. **Comparación de ontologías.** Con la finalidad de obtener las ontologías base que servirán para generar el modelo ontológico, se utilizó los criterios de evaluación abordados en un trabajo similar [41]. Dichos criterios están relacionados con el alcance y poseen un propósito similar al que pudiesen llegar a tener las posibles ontologías a reusarse, así como el nivel de cumplimiento de los requerimientos funcionales y no funcionales.

Para calificar cada uno de estos se usó un rango de valores cualitativos en donde:

- **Si-Totalmente (Si-T).** Indica que la ontología candidata cumple de manera exacta con el criterio calificado.

⁶<http://ws.nju.edu.cn/falcons/ontologysearch/>

⁷<http://www.sindice.com/>

⁸Weapon Ontology Annotation Using Boundary Describing Sequences. Ver mas en [7]

⁹<http://dbpedia.org/ontology/>

¹⁰<https://iptc.org/standards/newscodes/>

¹¹<http://nerd.eurecom.fr/ontology/>

¹²<https://wordnet.princeton.edu/>

- **Si-Parcialmente (Si-P).** Indica que la ontología candidata cumple de manera parcial con el criterio calificado.
- **No (N).** Indica que la ontología candidata no cumple con el criterio calificado.
- **Desconocido (D).** Indica que la ontología candidata no tiene documentación suficiente para determinar si es válida o no para ser reutilizada dentro del criterio calificado.

Los resultados de este análisis se presentan en la Tabla 4.5.

Criterio	Rango de Valores	Ontologías Candidatas				
		Weapon Ontology	DBpedia Ontology	IPTC News Codes	NERD	WordNet
Alcance Similar	[Si-T, Si-P, N, D]	Si-P	Si-P	N	N	N
Propósito Similar	[Si-T, Si-P, N, D]	Si-T	Si-P	N	N	N
Requerimientos no funcionales cubiertos	[Si-T, Si-P, N, D]	Si-P	Si-P	N	N	N
Requerimientos funcionales cubiertos	[Si-T, Si-P, N, D]	Si-P	SI-T	N	N	Si-P

Tabla 4.5: Cuadro comparativo de Ontologías.

Para determinar las ontologías no adecuadas se utilizó la misma heurística aplicada en [41], donde se afirma que si el desarrollador contestó “N” a los criterios “Alcance similar” y/o “Propósito similar” y/o “Requerimientos Funcionales cubiertos”, entonces la ontología es considerada no útil, y por tanto fue debe ser eliminada del conjunto de ontologías candidatas. Es así que, tanto IPTC News Codes como NERD han sido eliminadas del conjunto de ontologías candidatas.

3. **Selección de las ontologías.** En esta etapa se realizó la selección de la ontología más adecuada para ser reutilizada en el modelo ontológico a crear. Para tal finalidad se utilizó la Tabla propuesta en [42], donde se plantean cuatro dimensiones de evaluación relacionadas directamente con las características no funcionales de las ontologías. Estas dimensiones son:

- **Costo de Reutilización.** Esta dimensión hace referencia a la estimación del costo (económico y temporal) necesario para el reuso de la ontología candidata.

- **Esfuerzo de Comprensión.** Se refiere a la estimación del esfuerzo necesario para la compresión de la ontología candidata.
- **Esfuerzo de Integración.** Hace referencia a la estimación del esfuerzo necesario para la integración de la ontología candidata dentro de la ontología que está en proceso de desarrollo.
- **Confiabilidad.** Implica al análisis de si es o no pertinente confiar en la ontología seleccionada para ser reutilizada.

Cabe recalcar que cada una de las dimensiones tiene sus propios criterios de análisis, que pueden ser profundizados en [42]. Aquí se observa que es posible obtener una calificación para cada ontología candidata. Cada uno de los criterios se divide en un rango de valores lingüísticos, una explicación de cómo medir dicho criterio y un valor numérico. En la Tabla 4.6 se observan los resultados obtenidos.

Criterio	Peso	Valores		
		Weapon Ontology	DBPedia Ontology	WorNet
Costo de Reutilización				
Costo Económico de Reúso	(-)9	1	1	1
Tiempo Requerido para el Reúso	(-)7	1	2	1
Esfuerzo de Comprensión				
Calidad Documentación	(+)8	2	3	2
Disponibilidad de Conocimiento Externo	(+)7	1	3	2
Claridad del Código	(+)8	1	3	2
Esfuerzos de Integración				
Adecuación de extracción de conocimiento	(+)9	3	2	1
Adecuación de convenciones de nombrado	(+)5	3	2	1
Adecuación del lenguaje de implementación	(+)8	3	2	1
Conflictos de conocimiento	(-)8	1	2	1
Adaptación al razonador	(+)5	3	2	0
Necesidad de términos puente	(-)7	1	1	0
Confiabilidad				
Disponibilidad de pruebas	(+)8	1	3	0
Soporte Teórico	(+)8	2	3	1
Reputación del Equipo de Desarrollo	(+)8	3	3	3
Fiabilidad de Documentación	(+)3	3	3	2
Apoyo Práctico	(+)6	1	3	2
	TOTAL	1.11	1.19	0.75

*desconocido=0, bajo=1, medio=2 y alto=3

Tabla 4.6: Tabla de calificación de ontologías candidatas.

Una vez asignados los valores por parte del equipo de trabajo se procedió a calcular el puntaje total correspondiente a cada ontología candidata. Como se observa en la Tabla 4.6 existen pesos denotados con símbolo positivo y negativo, dichos pesos deben ser tratados de manera independiente para el cálculo del puntaje total, cuya fórmula es:

$$Puntaje_i = Puntaje_{i(+)} - Puntaje_{i(-)} \quad (4.1)$$

En donde:

$$Puntaje_{i(+)} = \sum_{j(+)} Valor_{T_{ij}} x \frac{Peso_j}{\sum_j Peso_j} \quad (4.2)$$

$$Puntaje_{i(-)} = \sum_{j(-)} Valor_{T_{ij}} x \frac{Peso_j}{\sum_j Peso_j} \quad (4.3)$$

Siendo :

- **Puntaje (+).** Puntaje para la ontología candidata i para el conjunto de criterios ponderados con $(+)$.
- **Puntaje (-).** Puntaje para la ontología candidata i para el conjunto de criterios ponderados con $(-)$.
- **i .** Ontología candidata particular.
- **j .** Criterio particular de las incluidas en la Tabla 4.6, $j(+)$ representa los criterios con peso positivo y $j(-)$ los criterios con peso negativo.
- **$Valor_{T_{ij}}$** Valor para el criterio de j en la ontología.
- **$Peso(j)$.** Peso numérico asociado al criterio j .

Como resultado final de esta etapa se determinó que DBpedia Ontology con 1.19 es la ontología con mayor puntaje, por lo que será utilizada en el proceso de creación de la ontología de armas de fuego. Además, se seleccionó Weapon Ontology [16]; que pese a ser un modelo ontológico conceptual, es el único modelo formal de armas de fuego que existe y que posee una clasificación de las mismas; por lo que sirve de base para la creación del modelo ontológico en cuestión.

4. **Personalización e integración de ontologías seleccionadas.** En esta etapa se realizó la personalización e integración de la ontología Weapon

Ontology con la DBpedia Ontology. Previamente, es importante recalcar ciertas especificaciones importantes para la generación del modelo ontológico de armas de fuego:

- I El modelo ontológico conceptual propuesto por Weapon Ontology fue utilizado como base inicial para el desarrollo de la ontología, donde se observa que la clasificación de armas se realiza de acuerdo a su mecanismo de disparo. Esta clasificación es generalmente aceptada [43] y se ajusta a los fines de este proyecto de tesis. Por esto, se realizó un proceso manual de creación de clases, relaciones y propiedades en Protege¹³, adicionando las características necesarias para cumplir con las preguntas de competencia.
- II De acuerdo a la información proporcionada por recursos ontológicos y no ontológicos; DBpedia Ontology e IMFDB¹⁴ respectivamente, se concluyó que se debían agregar más categorías a la clasificación original de armas de fuego presentada por Weapon Ontology, con la finalidad mejorar el modelo ontológico a crearse.
- III Se generó una propiedad llamada “source” con el fin de vincular a las instancias de la ontología con las instancias de DBpedia Ontology y enriquecer la información que contendrá dicho modelo ontológico.

La Figura 4.2 muestra se muestra la estructura de la ontología de armas de fuego, en donde se observa que consta de 20 clases y 7 propiedades de los datos. Además cabe mencionar que el *Uniform Resource Identifier* (URI) bajo el cual será publicado este modelo ontológico es www.semanticweb.org/firearms.

4.4. Desarrollo de un prototipo para la población de la ontología

Una vez creada la estructura de la ontología de armas de fuego es necesario poblarla con datos que la hagan usable para los propósitos de esta tesis. DBpedia por su parte, almacena en su ontología cientos de instancias pertenecientes a la clase Weapon¹⁵, las cuales se adaptan adecuadamente al modelo ontológico propuesto. Es así que el grupo de trabajo decidió realizar un mapeo entre los

¹³<http://protege.stanford.edu/>

¹⁴http://www.imfdb.org/wiki/Main_Page

¹⁵<http://dbpedia.org/ontology/Weapon>

diferentes tipos de armas existentes, de manera que estos se ajusten a las clases de la ontología creada.

Este proceso fue necesario para poblar la ontología con información de DBpedia, pero filtrando solamente aquella que se encuentra dentro del campo de estudio de esta tesis. Los criterios utilizados fueron: similitud de nombre, similitud de propiedades y decisión propia del grupo de trabajo en base a información encontrada en fuentes externas confiables como por ejemplo IMFDB, en donde se puede acceder a información de armas de fuego presentadas en películas.

Una vez hecho esto se procedió a desarrollar un módulo en lenguaje Java¹⁶ que permita poblar automáticamente la ontología de armas de fuego con datos extraídos de DBpedia. El resultado de este proceso será cargado en el servidor Virtuoso¹⁷, de manera que pueda ser utilizado en el siguiente capítulo tanto para anotación de videos como para el módulo de consulta.

A continuación se describen las entradas y salidas, requisitos, herramientas y finalmente el desarrollo de dicho módulo.

4.4.1. Entradas y salidas

- El módulo de población de la ontología recibirá como entradas las instancias de armas obtenidas de DBpedia así como el modelo ontológico en formato RDF.
- Se obtendrá como salida un archivo que contenga el modelo ontológico e instancias en formato RDF.

4.4.2. Herramientas de desarrollo

Para realizar la tarea de programación del prototipo de población de la ontología se utilizaron las herramientas que se describen brevemente a continuación.

Java¹⁸

Es lenguaje de programación orientado a objetos y de propósito general. Es utilizado ampliamente por los desarrolladores debido a que permite la portabilidad de sus aplicaciones. Entiéndase por portabilidad a la capacidad de ejecutar cualquier aplicación en cualquier sistema operativo sin importar donde se encuentre alojado, siempre y cuando dicha máquina tenga instalado Java. Existen

¹⁶<https://www.java.com/es/>

¹⁷<http://virtuoso.openlinksw.com/>

¹⁸<https://www.java.com/es/>

múltiples IDE que manejan este lenguaje de programación tales como Eclipse¹⁹ o Netbeans²⁰.

Apache Jena²¹

Es un framework gratuito de código abierto para Java que permite la construcción de aplicaciones que implementan tecnologías semánticas. Entre sus características principales provee un *Application Programming Interface* (API) para crear y leer grafos RDF, provee serialización de RDF a varios formatos como RDF/XML, Turtle²² o N3²³; persistencia de triplas, motor de consultas SPARQL, soporte para OWL, entre otras ventajas que lo hacen ideal para el desarrollo de proyectos que impliquen el uso de ontologías.

Previo al desarrollo del módulo, se descargó e integró las librerías de Apache Jena v 2.13.0 en el entorno de desarrollo NetBeans 8.0.1.

4.4.3. Desarrollo

El desarrollo del módulo para la población de la ontología se puede resumir en las siguientes etapas:

1. Consultar a DBpedia mediante SPARQL para obtener los datos necesarios de las instancias pertenecientes a la clase Weapon. En el Listing 4.1 se muestra la consulta utilizada para obtener el URI, nombre, peso, largo, calibre, fabricante, lugar de origen y categoría de las armas almacenadas en DBpedia.
2. Leer el modelo ontológico a ser poblado.
3. Crear nodos para cada propiedad obtenida de DBpedia.
4. Crear un recurso que representará al arma.
5. Asignar a cada propiedad del modelo ontológico el nodo que le corresponde.
6. Asignar la propiedad “categoría” correspondiente.
7. Asignar la propiedad “source” de manera que constituya un vínculo entre la ontología creada y la instancia alojada en DBpedia, que puede ampliar significativamente la información acerca de un arma en caso de ser necesario.

¹⁹<https://eclipse.org/>

²⁰<https://netbeans.org/>

²¹<https://jena.apache.org/>

²²<http://www.w3.org/TeamSubmission/turtle/>

²³<http://www.w3.org/TeamSubmission/n3/>

8. Escribir el resultado en un archivo en formato RDF.
9. Repetir desde el numeral 3 para cada uno de los registros obtenidos de DBpedia.

```
PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns/#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema/#>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
PREFIX dbpprop: <http://dbpedia.org/property/>
SELECT * WHERE{
    ?link  rdf:type  <http://dbpedia.org/ontology/Weapon> .
    ?link  rdfs:label  ?label .
    ?link  dbpedia-owl:weight  ?peso .
    ?link  dbpedia-owl:length  ?largo .
    ?link  dbpprop:cartridge  ?calibre .
    ?link  dbpprop:origin  ?origen .
    ?link  dbpprop:manufacturer  ?fabricante .
    ?link  dbpedia-owl:type  ?categoria .
    FILTER (LANG(?label) = 'en') }
}
```

Segmento de código 4.1: Consulta SPARQL para obtener las armas almacenadas en DBpedia.

Al finalizar el proceso se podrá observar en la Interfaz creada tanto los registros procesados de DBpedia, así como el archivo resultante, como se presenta en la Figura 4.2.

Una vez obtenido el archivo con la ontología de armas de fuego debidamente poblada se procedió a subirla al servidor Virtuoso. La información se encuentra ahora disponible para consultas mediante SPARQL. Esto se utilizará para el módulo de anotación de videos multimedia tratado en el siguiente capítulo. Finalmente, la ontología de armas de fuego y el módulo de población de la misma constituyen la prueba del cumplimiento de los objetivos planteados para este capítulo.

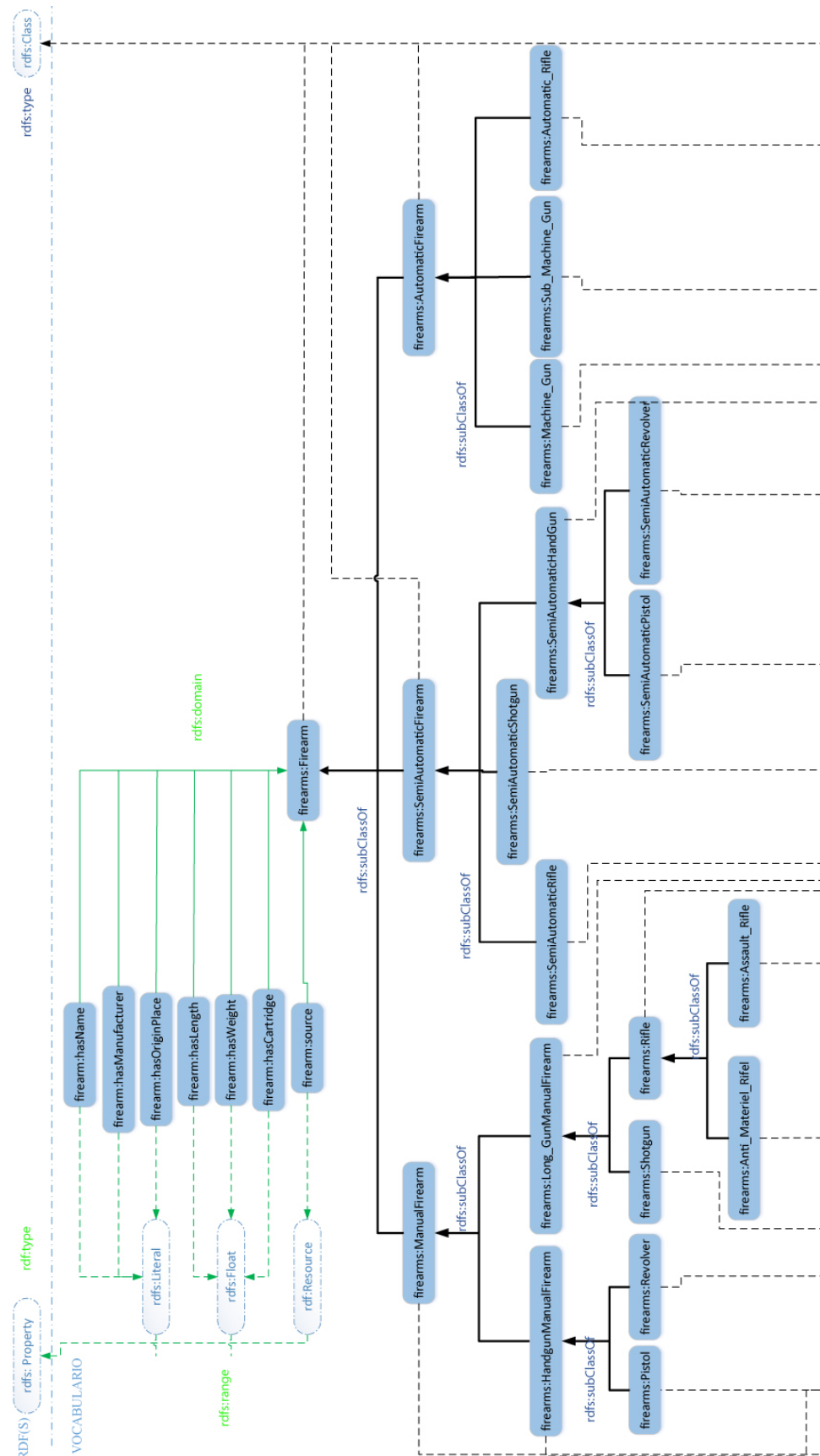


Figura 4.2: Modelo ontológico de clasificación de las armas de fuego.

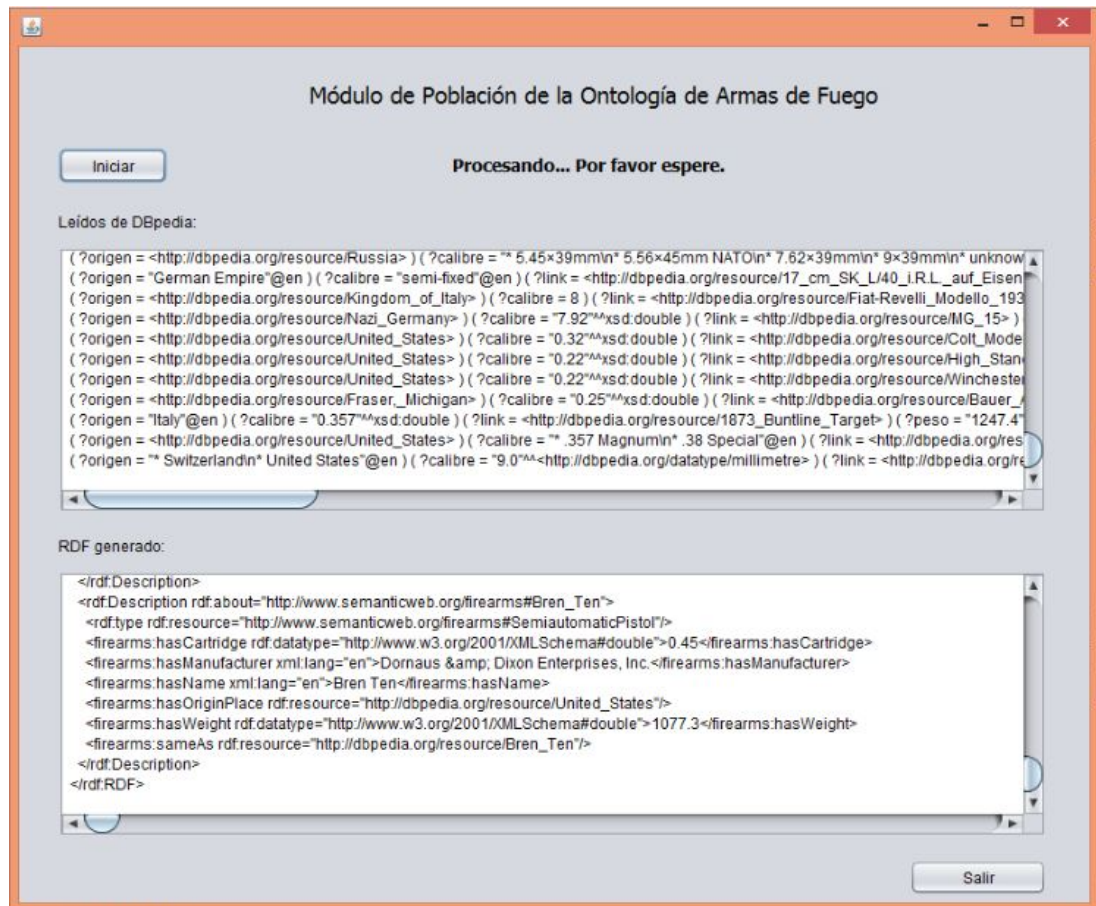


Figura 4.3: Resultado de la ejecución del módulo de población

Capítulo 5

Anotación Semántica de Contenido Televisivo

El uso de las tecnologías semánticas ha crecido notablemente durante los últimos años debido al constante aumento de la cantidad de información existente en la web. Dichas tecnologías permiten dar un significado a la información de tal forma que el computador pueda “comprender” el contenido y sea capaz de presentar los resultados esperados. Dentro de este contexto, la anotación semántica permite anotar formalmente los datos de uno o varios dominios de distinta índole creando así relaciones entre ellos o agrupando conceptos con el objetivo de facilitar las tareas de acceso, extracción e interpretación de la información.

En este capítulo se toman los resultados obtenidos del proceso de reconocimiento de armas de fuego para almacenarlos mediante un modelo ontológico formal utilizando anotación semántica. Esto con el objetivo de presentar los resultados a los usuarios mediante un prototipo de consulta que facilite el acceso adecuado a la información almacenada en un repositorio semántico. Finalmente se presenta el proceso obtenido como resultado del desarrollo del proyecto de aplicación de tecnologías semánticas y técnicas de reconocimiento de objetos para la identificación de armas de fuego en video, con miras de que pueda ser aplicado en proyectos afines.

5.1. Modelado formal para la anotación semántica de videos

Como se explicó en el Capítulo II, el modelamiento de material multimedia mediante ontologías sirve para representar de manera formal alguna situación o hecho del mundo real en donde se usen este tipo de medios. Por tal motivo

en la presente sección se realizará el modelamiento de armas de fuego en video tomando como punto de partida la ontología para la clasificación de armas de fuego obtenida en el Capítulo IV.

Para hacer uso adecuado de dicha ontología en la anotación semántica de videos, inicialmente se debe buscar un modelo ontológico que cumpla con esta finalidad y que se ajuste a los objetivos específicos del presente proyecto de tesis. Dicho modelo ontológico debe estar relacionado con el manejo de información televisiva, de tal forma que se pueda anotar datos importantes relacionados a los videos tales como duración, sinopsis, entre otros.

Para tal fin se investigó en Internet sobre ontologías que modelen programación televisiva. Además, se consultó algunos proyectos de tesis relacionados al estudio y desarrollo de aplicaciones para televisión digital [41]. En base a esto, se obtuvo como resultado dos ontologías que podrían cumplir con los requisitos necesarios para la anotación de videos, las mismas que son Programmes Ontology¹ de la BBC² y LinkedTV Ontology³ desarrollada para el proyecto Europeo de televisión LinkedTV FP7⁴.

Con el objetivo de determinar cuál de estas ontologías es la que mejor se adapta al presente proyecto de tesis se tomó en cuenta la Tabla 4.6 utilizada en el Capítulo IV para la selección de ontologías, en donde se presentan ciertos parámetros que deben ser evaluados. El resultado del análisis de la selección de la mejor ontología para la anotación de videos en base a las dos ontologías presentadas anteriormente se muestra en la Tabla 5.1., donde se observa que la ontología con mayor puntuación es la Programmes, por lo que será utilizada para el proceso de anotación de video que se desarrollará en este capítulo.

La Figura 5.1 muestra el modelo ontológico que propone la Ontología Programmes, el cual a simple vista presenta un modelo bastante completo ya que su granularidad llega inclusive a manejar intervalos de tiempo, los cuales son necesarios para desarrollar la fase final de este proyecto de tesis.

Con el fin de vincular la ontología de armas de fuego obtenida en el Capítulo IV con la ontología Programmes para la anotación de videos, se creó la propiedad ‘hasFirearm’ que relaciona una arma de fuego con la escena del video en la que aparece. Como se observa en la Figura 5.2, la ontología Programmes tiene una clase denominada ‘Scene’ la cual tiene una subclase denominada ‘ViolentScene’.

¹www.bbc.co.uk/ontologies/po

²www.bbc.co.uk/mundo

³www.semantics.eurecom.fr/linkedtv/

⁴www.linkedtv.eu

Criterio	Peso	Valores	
		Programmes	LinkedTV
Costo de Reutilización			
Costo Económico de Reúso	(-)9	1	1
Tiempo Requerido para el Reúso	(-)7	2	3
Esfuerzo de Comprensión			
Calidad Documentación	(+)8	3	2
Disponibilidad de Conocimiento Externo	(+)7	2	2
Claridad del Código	(+)8	3	2
Esfuerzos de Integración			
Adecuación de extracción de conocimiento	(+)9	2	1
Adecuación de convenciones de nombrado	(+)5	3	2
Adecuación del lenguaje de implementación	(+)8	3	3
Conflictos de conocimiento	(-)8	1	2
Adaptación al razonador	(+)5	2	1
Necesidad de términos puente	(-)7	1	2
Confiabilidad			
Disponibilidad de pruebas	(+)8	3	3
Soporte Teórico	(+)8	3	3
Reputación del Equipo de Desarrollo	(+)8	3	3
Fiabilidad de Documentación	(+)3	3	3
Apoyo Práctico	(+)6	2	2
	Total	1.45	0.32

*desconocido=0, bajo=1, medio=2 y alto=3

Tabla 5.1: Selección de ontología a reutilizar.

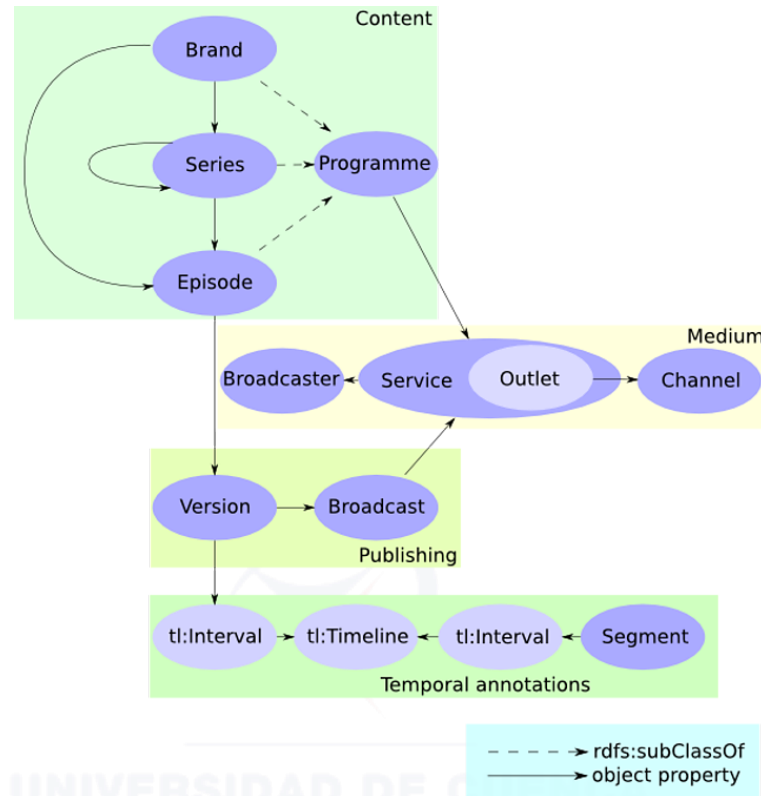


Figura 5.1: Ontología Programmes [8].

Esta clase será la base para realizar la anotación semántica de los videos y constará de las siguientes propiedades:

- **hasFirearm.** Propiedad de enlace entre las dos ontologías.
- **hasVideo.** Indica el video en el que aparece el arma especificada mediante **hasFirearm** en una escena específica.
- **beginsAtInt.** Tiempo inicial del intervalo de aparición del arma.
- **endsAtInt.** Tiempo final del intervalo de aparición del arma.

Una vez definido el metamodelo de la ontología para la anotación de videos, en la siguiente sección se procederá a explicar el proceso de anotación mediante la creación de un módulo que permita realizar esta tarea.

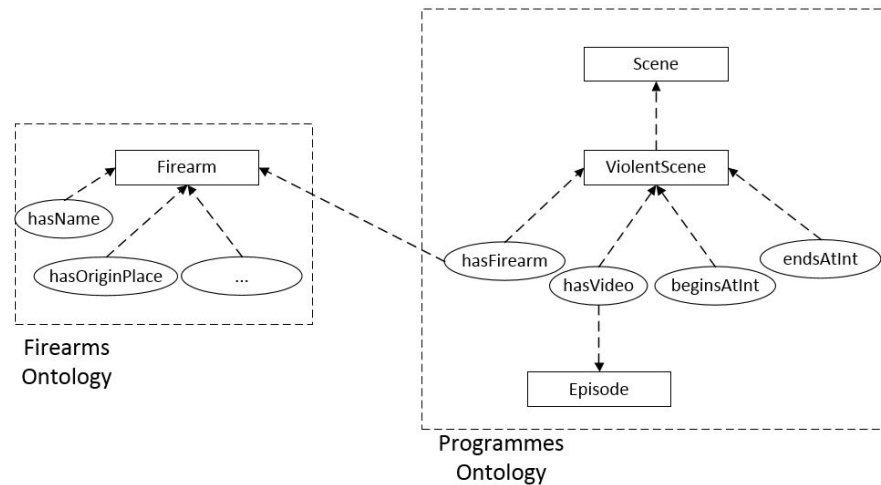


Figura 5.2: Metamodelo de la Ontología para anotación de videos.

5.2. Desarrollo de un prototipo para la anotación semántica de videos

En el Capítulo III se desarrolló un prototipo para reconocer armas de fuego presentes en un video, de donde se obtuvo un archivo XML que contiene los resultados de dicho proceso. Además, se generó una ontología para representar formalmente armas de fuego, misma que fue vinculada con una ontología que representase formalmente distintos tipos de contenido multimedia. De dicha unión se obtuvo una ontología que será utilizada para el proceso de anotación semántica de videos (Sección 5.1), que no es otra cosa que poblar la ontología con datos que representen la relación existente entre los videos y los objetos encontrados en ellos, con las debidas especificaciones de tiempos de aparición. Finalmente, fue necesario desarrollar un prototipo para realizar la anotación semántica de los videos tomando en cuenta los elementos mencionados.

El módulo de anotación semántica de videos se desarrolló en lenguaje Java. Cuenta con una interfaz que permite mostrar las ocurrencias de armas en los videos y su factor de similitud comparadas con la ontología de armas de fuego. El proceso se realiza de forma automática, almacenando todas las ocurrencias, o de forma manual, almacenando solamente las seleccionadas por el usuario. La ontología poblada resultante del proceso de anotación será cargada en el servidor Virtuoso. Dicha ontología constituirá la base para el módulo de consulta tratado más adelante, pues contendrá conjuntamente con la ontología de armas de fuego, toda la información acerca de las armas, videos, y las apariciones de las armas en

dichos videos.

5.2.1. Entradas y salidas

A continuación se listan las entradas y salidas correspondientes al prototipo de anotación semántica.

Entradas

- Archivo .xml que contiene las armas de fuego reconocida
- Ontología de armas de fuego poblada en formato RDF
- Ontología destinada a la anotación en formato RDF
- Base de datos de videos

Salida

- Ontología poblada con videos y anotaciones en formato RDF

5.2.2. Requisitos previos a la programación

Como se mencionó en secciones anteriores es necesario contar con librerías que faciliten el manejo de ontologías en lenguaje Java. Previo al desarrollo del módulo, se descargó e integró las librerías de Apache Jena v 2.13.0 en el entorno de desarrollo NetBeans 8.0.1.

5.2.3. Desarrollo

A continuación se describe el procedimiento llevado a cabo para desarrollar el módulo para la anotación semántica de videos.

1. Leer el archivo XML que contiene las armas de fuego detectadas.
2. Construir una clase que permita representar y almacenar en memoria las armas de fuego como objetos.
3. Comparar cada objeto encontrado con las instancias de la ontología de armas de fuego para obtener su factor de similitud.

Para esta fase del proceso se decidió implementar lo que se conoce como la distancia de Jaro Winkler [44], que es una medida de la similitud entre dos cadenas de texto. Una vez implementadas las funciones propias de

esta clase, estas permitirán ingresar dos cadenas de texto para devolver el factor de similitud entre 0 y 1, indicando ninguna similitud e igualdad, respectivamente.

Los resultados obtenidos se mostrarán en una tabla que consta de: en primer lugar, el registro analizado, luego el video en donde se reconocieron las armas, el arma reconocida, el inicio y fin de la ocurrencia, la instancia de la ontología de armas de fuego con la que se produce la similitud y finalmente el factor de similitud, tal como se muestra en la Figura 5.3. De esta forma el usuario podrá seleccionar un proceso automático para anotar todas las ocurrencias, o manual para almacenar solamente las seleccionadas.

No. Registro	Id Video	Id Arma	Nombre	Inicio	Fin	Instancia	Factor de coincidencia
1	2	3	Mossberg 590 Mariner	2920.0	9560.0	Mossberg 590 Mariner	1.0
2	2	3	Mossberg 590 Mariner	14600.0	19800.0	Mossberg 590 Mariner	1.0
3	2	3	Mossberg 590 Mariner	23560.0	28400.0	Mossberg 590 Mariner	1.0

Figura 5.3: Resultados del emparejamiento de armas detectadas y registros en la ontología

4. Para continuar con el proceso de anotación, es necesario leer la ontología destinada para las anotaciones y generar un modelo ontológico en memoria.
5. Agregar cada registro de la tabla presentada en la Figura 5.3 en el modelo de la ontología para anotaciones. Tomando en cuenta las entidades disponibles en la ontología, se decidió almacenar cada registro como una escena violenta.

Un ejemplo del formato utilizado se observa en el Listing 5.1, en donde la escena violenta es de tipo “<http://purl.org/dc/elements/1.1/Scene>” y “<http://purl.org/dc/elements/1.1/ViolentScene>”, la propiedad `hasFirearm` indicará el arma que aparece en el video, la propiedad `hasVideo` indicará el video en el que aparece el arma, por último, `beginAtInt` y `endsAtInt`, indican respectivamente el tiempo de inicio y fin de la aparición del arma.

6. Una vez finalizado el proceso de anotación se almacenará el modelo ontológico de vuelta en el archivo RDF, siendo este el resultado final esperado de la ejecución del módulo.


```
<rdf:Description
  rdf:about="http://purl.org/dc/elements/1.1/Escena_Violenta_1" >
  <rdf:type rdf:resource="http://purl.org/dc/elements/1.1/Scene" />
  <rdf:type
    rdf:resource="http://purl.org/NET/c4dm/event.owl#Event" />
  <rdf:type
    rdf:resource="http://purl.org/dc/elements/1.1/ViolentScene" />
  <dc:hasFirearm
    rdf:resource="http://www.semanticweb.org/firearms#Mossberg_590
_Mariner" />
  <dc:hasVideo
    rdf:resource="http://purl.org/ontology/po/Episode/video_de_prueba
_2_(escopeta)" />
  <tl:beginsAtInt
    rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >2920
  </tl:beginsAtInt>
  <tl:endsAtInt
    rdf:datatype="http://www.w3.org/2001/XMLSchema#int" >9560
  </tl:endsAtInt>
</rdf:Description>
```

Segmento de código 5.1: Formato utilizado para registro de escenas violentas.

Habiendo obtenido el archivo de anotaciones con todos los registros de escenas violentas, es decir, anotaciones semánticas, se procede a subir este archivo al servidor Virtuoso para su posterior consulta. Con esto, se da por cumplido el objetivo de realizar la anotación semántica de armas de fuego en videos. En la siguiente sección se describirá el desarrollo del prototipo de consulta de la información almacenada en el servidor.

5.3. Desarrollo de un prototipo para la consulta a un repositorio semántico

Del proceso de anotación realizado en la sección anterior se obtuvo un modelo ontológico poblado con la información de armas de fuego existentes en videos el cual se encuentra almacenado en un repositorio semántico que provee el Servidor Virtuoso. La forma de extraer información de este repositorio es mediante el

uso del lenguaje de consulta SPARQL tratado en el Capítulo II. Para este fin, Virtuoso permite acceder a los datos ya sea directamente desde su consola de generación de consultas SPARQL o mediante peticiones HTTP directamente al servidor. Entiéndase por petición HTTP a una solicitud que se realiza al servidor, en este caso Virtuoso, para obtener información en algún formato específico.

En la presente sección se ha puesto principal énfasis en la creación de un módulo que permita realizar las tareas de consulta y que pueda ser usado tanto por usuarios regulares así como por usuarios que tienen conocimiento del lenguaje de consulta SPARQL y que desearan consultar información adicional. En la Figura 5.4 se presenta un esquema representativo del módulo de consulta en donde se observa que el repositorio semántico se encuentra dentro del servidor (Virtuoso Server) al cual se accede mediante un módulo web que emite peticiones *Hypertext Transfer Protocol* (HTTP) y recibe información en formato *JavaScript Object Notation* (JSON) para mostrarse sobre la interfaz de usuario.

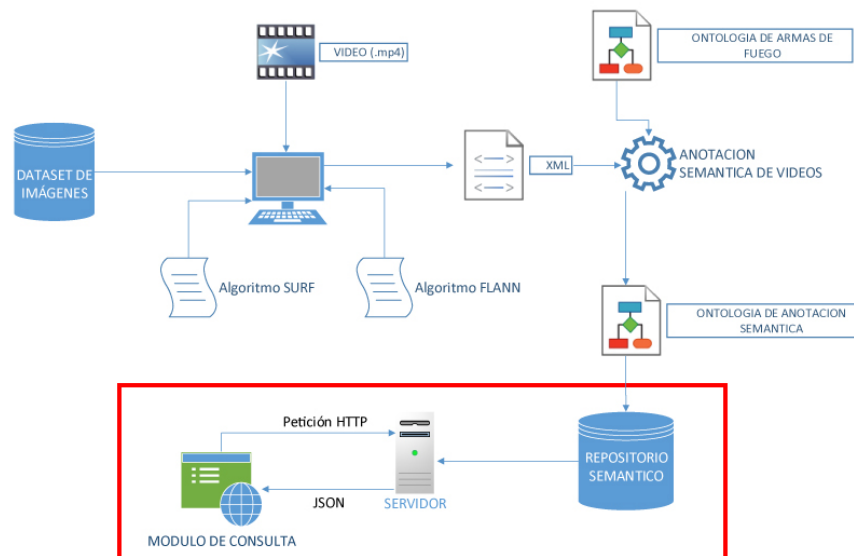


Figura 5.4: En rojo, Módulo de Consulta al Repositorio Semántico

5.3.1. Entradas y salidas

Como se observa en la Figura 5.4, el módulo comprende las siguientes entradas y salidas:

Entradas

- **Repositorio Semántico.** En dicho repositorio se encuentra almacenada



la ontología que contiene la anotación de las armas de fuego en los videos analizados. Este repositorio se encuentra alojado en el servidor Virtuoso.

Salidas

- **Consultas SPARQL.** En base a las consultas que se realicen se obtendrá conjuntos de información en formato JSON, los mismos que serán formateados previamente a la presentación en la interfaz de usuario del módulo de consulta.

5.3.2. Herramientas de desarrollo

Para el desarrollo del módulo de consulta se utilizaron tecnologías web que permitan mantener al código fuente separado, es decir en un modelo por capas. A continuación se explica brevemente las tecnologías utilizadas.

HyperText Markup Language (HTML)

Es un lenguaje de marcado utilizado para la representación de contenido en Internet mediante páginas web. Dicho lenguaje utiliza etiquetas para representar los distintos elementos que forman la estructura de estas páginas. Actualmente este lenguaje se encuentra en su versión 5 y está siendo utilizada por los desarrolladores debido a que presta grandes facilidades al momento de agregar contenido multimedia sin la necesidad de agregar código innecesario, entre otras ventajas. En este proyecto de tesis se ha utilizado HTML5⁵ para la modelación de la interfaz de usuario del módulo de consulta.

JavaScript⁶

Es un lenguaje de programación muy utilizado en el desarrollo de páginas y aplicaciones web. Funciona del lado del cliente y trabaja en conjunto con HTML con la finalidad de presentar un contenido dinámico en las mismas. Actualmente se encuentra en su versión 1.8.5 y sirve de enlace entre la interfaz de usuario y la capa de negocio por lo que en el presente proyecto de tesis será utilizado para recibir peticiones desde la interfaz de usuario y redireccionarlas al servidor para obtener resultados, procesarlos y luego presentarlos en pantalla.

PHP Hypertext Pre-processor (PHP)⁷

Es un lenguaje de programación utilizado en el desarrollo web. Funciona del lado del servidor y puede ser usado con HTML para crear sitios web de manera dinámica. Actualmente se encuentra en su versión 5.6.7. Será utilizado para

⁵<http://www.w3.org/TR/html5/>

⁶<https://www.javascript.com/>

⁷<http://php.net/>

modelar la capa de datos ya que gracias a su soporte para conectividad con servidores y SGBDs permite acceder de manera fácil a los datos almacenados en cualquier repositorio, para este caso en específico, el acceso a Virtuoso.

Cascading Style Sheets (CSS)⁸

Es un lenguaje utilizado conjuntamente con HTML para dar estilo y formato a la estructura del sitio o aplicación web. Es de mucha utilidad ya que mantiene la estructura web como tal separada de la estética logrando así que cualquier cambio que se desee realizar sobre el estilo del sitio web no afecte a la estructura. Existen librerías predefinidas de estilos de las cuales se puede hacer uso y obtener un resultado más estético. Una de ellas es la denominada Bootstrap⁹ que permite manejar fácilmente el redimensionamiento automático del sitio web o aplicación dependiendo del dispositivo en el cual se presente el contenido.

5.3.3. Desarrollo

Habiendo definido cuáles serán las entradas y salidas del módulo así como las herramientas de desarrollo que se utilizarán para este fin, se procede a describir brevemente el proceso realizado para la creación del módulo de consulta.

1. **Diseño de la interfaz de usuario.** Para esta tarea se utilizó el lenguaje HTML5 conjuntamente con CSS para estilizar el módulo. Dicha interfaz incluirá las pestañas de Inicio, Consultas Predeterminadas, Consultas Personalizadas, SPARQL Endpoint y Acerca de, cuyas funcionalidades se detallarán más adelante.
2. **Diseño de la capa de negocio.** Para el desarrollo de esta capa se utilizó JavaScript generando varias funciones tanto para preguntas predefinidas así como para preguntas personalizadas. Todas las funciones creadas en esta capa permiten obtener información desde la capa de datos en base a las peticiones que se realicen desde la interfaz de usuario.
3. **Diseño de la capa de datos.** La capa de datos es la encargada de conectarse con el servidor que contiene al repositorio semántico y se encuentra programada en PHP. Dicha conexión se realiza mediante peticiones HTTP; como se indicó anteriormente, lo cual permite que se obtengan resultados en formato de datos JSON. Estos datos son utilizados en la capa de negocio para ser formateados y posteriormente mostrados en la interfaz de usuario.

⁸<http://www.w3.org/Style/CSS/>

⁹<http://getbootstrap.com/>

A continuación se detalla el conjunto de interfaces y funcionalidad con la que cuenta el módulo de consultas una vez finalizado su desarrollo:

Inicio. Contiene información relevante a la funcionalidad del módulo y cada una de sus secciones como se observa en la Figura 5.5.



Figura 5.5: Captura de la interfaz Inicio

Consultas Predefinidas. Sección destinada a mostrar los resultados de preguntas que respondan y den solución a inquietudes comunes que pudiesen tener los usuarios sobre la información almacenada en el repositorio. El equipo de desarrollo consideró varias inquietudes a las que este módulo podría responder mediante consultas. Como producto de esto se generaron las siguientes preguntas:

- Todas las escenas violentas almacenadas en el repositorio semántico.
- Todos los videos que contienen armas de fuego.
- Todos los videos que contienen escenas violentas.
- Todas las armas de fuego existentes en las escenas violentas.
- Todas las armas de tipo “Manual” existentes en los videos.
- Todas las armas de Tipo “Semiautomático” existentes en los videos.

- Todas las armas de Tipo “Automático” existentes en los videos.
- Todas las armas existentes en los videos con un enlace de información adicional.
- Número de ocurrencias de cada arma de fuego que aparece en los videos.

En la Figura 5.6 se muestran los resultados de la ejecución de la consulta predefinida número 5. Este es un ejemplo de la manera en la cual se presentan las distintas consultas al momento de su ejecución.

Consultas Predefinidas				
Consultas		Consulta 5 - Descripción		
Escenas Violentas Existentes.		Esta consulta muestra todas las armas de Tipo Manual existentes en los videos		
Videos que Contienen Armas de Fuego		Video	Escena	Inicio Fin Arma
Videos que Contienen Escenas Violentas		http://purl.org/ontology/po/Episode/video_de_prueba_1_(escopeta_giro)	http://purl.org/dc/elements/1.1/Escena_Violenta_7	31280 33320 http://www.semanticweb.org/firearms#Mossberg_590_Mariner
Armas de Fuego en Escenas Violentas		http://purl.org/ontology/po/Episode/video_de_prueba_1_(escopeta_giro)	http://purl.org/dc/elements/1.1/Escena_Violenta_4	2600 11000 http://www.semanticweb.org/firearms#Mossberg_590_Mariner
Armas de Tipo Manual Existentes en los Videos		http://purl.org/ontology/po/Episode/video_de_prueba_1_(escopeta_giro)	http://purl.org/dc/elements/1.1/Escena_Violenta_5	16400 20600 http://www.semanticweb.org/firearms#Mossberg_590_Mariner
Armas de Tipo Semiautomático Existentes en los Videos		http://purl.org/ontology/po/Episode/video_de_prueba_1_(escopeta_giro)	http://purl.org/dc/elements/1.1/Escena_Violenta_6	25960 29000 http://www.semanticweb.org/firearms#Mossberg_590_Mariner
Armas de Tipo Automático Existentes en los Videos		http://purl.org/ontology/po/Episode/video_de_prueba_1_(escopeta_giro)	http://purl.org/dc/elements/1.1/Escena_Violenta_1	2920 9560 http://www.semanticweb.org/firearms#Mossberg_590_Mariner
Armas existentes en Videos - Enlace Adicional				
Ocurrencias de las Armas de Fuego en los Videos				

Figura 5.6: Captura de la interfaz Consultas Predefinidas

Consultas Personalizadas. Sección destinada a mostrar resultados en base a ciertos parámetros que el usuario puede ingresar manualmente. Estos parámetros corresponden a características de las armas de fuego como nombre, lugar de origen, fabricante, longitud, peso y calibre. El usuario puede ingresar uno o varios de estos parámetros y se mostrará como resultado únicamente la información correspondiente a la combinación de los datos ingresados, tal como se observa en la Figura 5.7.

SPARQL Endpoint. Esta sección contiene un enlace directo al endpoint que provee Virtuoso para las consultas en lenguaje SPARQL. Se debe entender como *endpoint* a una interfaz a través de la cual se puede enviar y recibir información. Por lo tanto, el equipo de desarrollo encontró necesario colocar esta interfaz ya que de esta manera se da la posibilidad de que usuarios expertos en el manejo de lenguaje SPARQL puedan escribir sus propias consultas y obtener resultados más personalizados. Un ejemplo de esto se presenta en la Figura 5.8.

Consultas Personalizadas

A continuación se presentan varios campos de texto.

- Cada uno de ellos corresponde a diferentes propiedades que tienen las armas de fuego anotadas de manera semántica en videos.
- Usted puede ingresar el valor o combinación de valores que desee y presionar sobre el boton 'Consultar'.
- Si dicha combinación de valores corresponde a alguna(s) coincidencia(s) exacta(s) dentro del repositorio semántico se devolverá resultados.

Nombre	Dan Wesson	Lugar de Origen	Ejm: Rusia	Fabricante	Ejm: Kalishnikov	Peso	Ejm: 12
Longitud	Ejm: 25	Calibre	Ejm: 9	Consultar			

Video De Prueba 5 (revolver Giro)

Video URI: [http://purl.org/ontology/po/Episode/video_de_prueba_5_\(revolver_giro\)](http://purl.org/ontology/po/Episode/video_de_prueba_5_(revolver_giro))

Sinopsis: Video con revolver CO2 Dan Wesson 6" con giro

Escena: http://purl.org/dc/elements/1.1/Escena_Violenta_18

Intervalo de Aparición de la Arma "Dan Wesson"

Inicio: 991 ms. **Fin:** 19404 ms.

Arma de Fuego URI: http://www.semanticweb.org/firearms#Dan_Wesson_Model_27

Nombre: Dan Wesson

Origen: United States

Fabricante: http://dbpedia.org/resource/Smith_&_amp_Wesson

Longitud: 0.193675

Peso: 1065.96



Figura 5.7: Captura de la interfaz Consultas Personalizadas

Acerca de. En esta sección se presenta información adicional como el proyecto de tesis, el equipo de desarrollo y el director.

5.4. Proceso propuesto para reconocimiento y anotación semántica de objetos en video

El proceso de reconocimiento de armas de fuego en video y anotación semántica de videos concluye con el módulo de consulta tratado en la sección anterior. Como resultado de todo el trabajo llevado a cabo en este proyecto de tesis, se obtuvo un proceso definido a seguir, el mismo que se propone de manera general para desarrollar tareas similares.

En el esquema de la Figura 5.9 se observa el procedimiento seguido para todo este proyecto. Al inicio se cuenta con una base de datos de imágenes y datos que representan las armas de fuego a reconocer. Por otro lado están los videos en los que se sabe existen armas de fuego que necesitan ser detectadas y reconocidas. Entonces se realiza un proceso computacional aplicando algoritmos de reconocimiento de imágenes como SURF y FLANN, produciendo como resultado un archivo XML que contiene registros de las armas de fuego y los videos en los que fueron encontradas. El siguiente paso consiste en la anotación semántica de los

SPARQL

[Volver](#) [Ver Modelo Ontológico de Armas de Fuego](#) [Ver Metamodelo de Anotación de Videos](#)

Virtuoso SPARQL Query Editor

[About](#) | [Namespace Prefixes](#) | [Inference rules](#)

Default Data Set Name (Graph IRI)

Query Text

```
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX tl: <http://purl.org/NET/c4dm/timeline.owl#>
PREFIX firearms: <http://www.semanticweb.org/firearms#>
select distinct ?Arma, ?Nombre, ?Lugar_Origen, ?Fabricante, ?Peso, ?Longitud, ?Calibre
From <http://localhost:8890/ anotacion>
From <http://localhost:8890/firearms>
where
{
  ?SceneViolence dc:hasFirearm ?Arma .
  ?Arma firearms:hasName ?Nombre.
  ?Arma firearms:hasOriginPlace ?Lugar_Origen.
  ?Arma firearms:hasManufacturer ?Fabricante.
  ?Arma firearms:hasWeight ?Peso.
  ?Arma firearms:hasLength ?Longitud.
  ?Arma firearms:hasCartridge ?Calibre.
}
order by ?Arma
```

(Security restrictions of this server do not allow you to retrieve remote RDF data, see [details](#))

Results Format: HTML (The CXML output is disabled, see [details](#))

Figura 5.8: Captura de la interfaz SPARQL endpoint

videos. Para dicha tarea, se considera como entradas al archivo XML producido en el paso anterior, así como una ontología de armas de fuego. Se aplica entonces otro proceso computacional que permitirá generar las anotaciones semánticas en base a los datos de entrada, produciendo como salida el archivo de la ontología de anotación debidamente poblada. Finalmente, el archivo que contiene toda la información es colocado en un repositorio semántico, el cual por medio de un servidor y un módulo de consulta podrá ser accedido para obtener los reportes requeridos. De esta manera se ha descrito a breves rasgos todo el proceso realizado en este proyecto de tesis.

5.4.1. Prototipos propuestos

Para llevar a cabo todo el proceso descrito anteriormente, se desarrollaron tres prototipos o módulos principales y un módulo auxiliar descritos a continuación.

Prototipos principales

Prototipo de detección y reconocimiento de armas de fuego en video.

El objetivo de este prototipo es procesar un video detectando armas de fuego y reconociéndolas de acuerdo a una base de datos existente para producir una salida que describa los objetos encontrados.

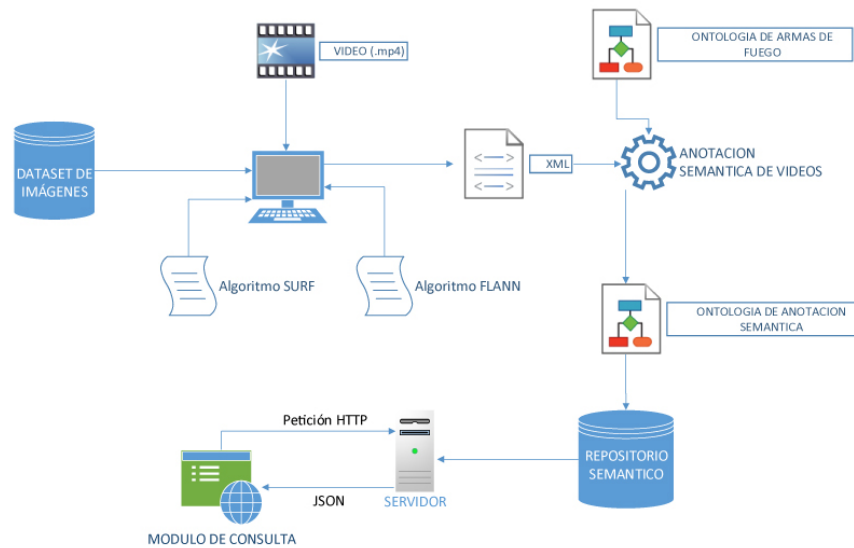


Figura 5.9: Esquema del proceso de reconocimiento y anotación de objetos en video

Prototipo de anotación semántica de videos. Tomando la salida del prototipo anterior, así como la ontología de armas de fuego y la ontología para anotaciones semánticas, el objetivo es registrar los datos producidos anteriormente en forma de anotaciones semánticas que modelen formalmente como escenas violentas la información acerca de videos, armas de fuego y sus intervalos de aparición.

Prototipo de consulta al repositorio semántico. Este prototipo tiene como propósito proveer un entorno amigable en el que el usuario pueda realizar consultas y visualizar reportes que presenten la información almacenada en el repositorio semántico. Para el caso de este proyecto se implementaron varios métodos de consulta que responden a requisitos establecidos previamente, sin embargo esto dependerá del contexto para el cual se desarrolle un proyecto de este tipo.

Prototipo auxiliar

Prototipo de población de ontología de armas de fuego. Se requería poblar la ontología de armas de fuego tomando datos de DBpedia, por lo que este prototipo lee datos de dicha fuente y los transforma para encajar en el modelo ontológico de armas de fuego, produciendo como resultado la ontología de armas de fuego debidamente poblada.

Para el caso de otro proyecto similar, se debe analizar previamente la exis-

tencia de ontologías que modelen los objetos, escenarios o situaciones tratadas; buscar en primer lugar reusar recursos disponibles y finalmente; si no se ha encontrado recursos útiles, entonces sí generar un nuevo modelo ontológico.

Además, es probable que sea necesario crear otros módulos auxiliares para realizar ciertas tareas, como por ejemplo depuración de datos, población de ontologías, comunicación entre distintas fuentes de información, etc.

Tal como se explicó en la Figura 5.9, producto de este proyecto de tesis se ha obtenido un proceso definido que puede ser aplicado para cualquier proyecto relacionado con reconocimiento de objetos, su modelado formal y la anotación semántica de los objetos. De la experiencia con este proyecto de tesis, se ha decidido dividir dicho proceso en tres fases, en donde cada una cumple con una tarea específica y constituye un requisito para la siguiente. A continuación se detalla cada una de ellas.

5.4.2. Fases propuestas

Fase I – Detección y reconocimiento de objetos

Una vez se ha definido el proyecto a desarrollar, la primera fase consiste en definir los objetos a ser analizados. Es importante determinar las características de lo que se pretende detectar y/o reconocer, pues de ello dependerá la selección de los algoritmos más adecuados.

El siguiente paso es la selección de algoritmos de detección y reconocimiento de objetos en video, pues existen distintas técnicas para dicha tarea, como se mencionó en el Capítulo II. En esta fase, se debe realizar el diseño del módulo de reconocimiento, para determinar el flujo del programa, las entradas, salidas y la manera en que se realizará el proceso de reconocimiento.

Finalmente, se desarrolla el módulo de detección y reconocimiento aplicando los algoritmos pertinentes para obtener los resultados deseados. La salida debe estar en un formato de datos fácilmente legible, pensando en que constituirá la entrada de otro módulo.

Fase II – Módulo de anotación semántica

Esta fase es quizá la más crucial del proyecto; consiste en definir el modelo formal, es decir, las ontologías a utilizarse para representar tanto los objetos tratados así como las anotaciones semánticas. Como se mencionó en el Capítulo II, la anotación consiste en agregar notas que permitan registrar información



agregada a un objeto. Una manera de realizar el proceso de anotación es realizar una anotación de tipo semántica, utilizando una ontología como modelo formal para representar la relación entre los objetos y la información que se desea anotar. Comúnmente la relación se define mediante una o varias propiedades que vinculan la clase o entidad que modela los objetos y aquella que permite modelar las anotaciones.

Para desarrollar el módulo de anotación, básicamente se tomarán los registros obtenidos en el módulo anterior a los cuales se les dará el formato adecuado registrando la información en el modelo ontológico. La salida será un archivo que contendrá las anotaciones semánticas de los videos procesados. Para dicho fin, se debe elegir un lenguaje de programación y librerías que trabajen bien en proyectos relacionados a las tecnologías semánticas. En este caso se utilizó Java como lenguaje de programación y las librerías de Jena para el manejo de las ontologías. El archivo resultante, será cargado a un repositorio para su posterior consulta por parte de los usuarios mediante un módulo diseñado para ello.

Fase III – Módulo de consulta

Generalmente el objetivo final de un proyecto es contar con información que presente los resultados obtenidos. Entonces, en esta tercera fase es recomendable desarrollar un módulo para la consulta de la información o brindar al usuario un entorno al que pueda acceder libremente. Para llevar a cabo esta tarea es necesario almacenar los datos en un repositorio ya sea local en un computador o en la web. Una vez hecho esto, se desarrollará el programa en un lenguaje de programación adecuado que permita cubrir los requerimientos establecidos en el proyecto. Se pueden presentar consultas predefinidas que satisfagan cuestionamientos planteados y/o proporcionar un *endpoint* en donde el usuario pueda construir por sí mismo sus consultas. De ser necesario se presentarán gráficos o información formateada en lenguaje natural para hacerla más amigable a quien la visualiza. Es así que, esta fase queda concluida al proporcionar los resultados de las dos fases previas de manera que sean fácilmente interpretables cumpliendo con el objetivo del proyecto que se esté llevando a cabo.

Para finalizar, es importante recalcar que estas fases pueden ser modificadas de acuerdo a los requerimientos del proyecto. Los módulos que se desarrollen pueden ser los nombrados anteriormente, pueden agregarse u omitirse otros de acuerdo a las necesidades del proyecto. Además, de ser pertinente podrían integrarse en uno solo que incluya las tres fases mencionadas anteriormente. El proceso descrito en



CAPÍTULO 5. ANOTACIÓN SEMÁNTICA DE CONTENIDO TELEVISIVO

esta sección engloba el trabajo realizado en este proyecto de tesis y se considera pertinente su aplicación en proyectos que pretendan realizar reconocimiento de objetos con miras a la anotación semántica de los mismos.



Capítulo 6

Conclusiones y Futuras Líneas de Investigación

Este capítulo está destinado a las conclusiones obtenidas como producto del desarrollo de este proyecto de tesis, las cuales se presentan de acuerdo a los objetivos específicos planteados inicialmente en el Capítulo 1. Además, se introduce a futuras líneas de investigación que puedan ser desarrolladas para aportar a la mejora tanto de los procedimientos como de los resultados obtenidos en cada fase de este proyecto.

6.1. Conclusiones

Para el desarrollo de este proyecto se inició con una investigación que permitió adquirir los conocimientos necesarios referentes a cada una de las áreas implicadas en el mismo. Dicho estudio comprendió temas relacionados a la detección y reconocimiento de objetos en video, creación y manejo de ontologías así como anotación semántica de datos. A partir de ello, el primer paso fue desarrollar un módulo de detección y reconocimiento de armas de fuego en videos. Por otra parte, se generó una ontología que modele la clasificación de armas de fuego, la cual fue posteriormente poblada y vinculada a una ontología que modela programación televisiva. Esto con el fin de obtener una ontología que permita realizar las anotaciones semánticas de las armas de fuego encontradas en los videos. Las anotaciones resultantes se almacenaron en un repositorio semántico con el fin de ser consultado por los usuarios para obtener información a través de un módulo de consulta. Finalmente se presenta todo el trabajo realizado en forma de un proceso que pueda aplicarse en proyectos afines.

En base a lo dicho anteriormente se puede concluir que se han cumplido de



manera exitosa los objetivos específicos planteados al inicio del proyecto, tal como se describe a continuación:

- **Realizar el reconocimiento de un conjunto determinado de armas de fuego en un video específico mediante la generación de un prototipo.**

Este objetivo se cumplió mediante el desarrollo de un módulo que se encarga de procesar un video detectando y reconociendo armas de fuego. Para llevarlo a cabo se realizó una investigación en la que se observó que la rama del procesamiento de imágenes y video es muy extensa, pudiendo dividirse en tres niveles que engloban desde un procesamiento general, hasta un análisis profundo de contenido implicando inferencia. La detección y reconocimiento de objetos en video se encuentra dentro del nivel intermedio, siendo la más desarrollada hoy en día. Así mismo, su constante evolución ha permitido su integración en distintos ámbitos tecnológicos como el militar, científico, comercial, investigativo, entre otros. Esto comprueba el gran potencial de este tipo de tecnologías para resolver problemas de distinta índole.

Durante el desarrollo de este proyecto fue posible analizar distintas técnicas de detección y reconocimiento, eligiendo aquellas que más se ajustaban a las necesidades del mismo. Se aplicaron algoritmos de detección de orillas y puntos clave para detectar objetos, así como un enfoque continuo para tratar el video como un conjunto de imágenes independientes. De la experiencia obtenida durante el desarrollo se recomienda hacer siempre una valoración previa antes de aplicar un algoritmo, así como basarse en documentación oficial para lograr mejores resultados. Finalmente, el módulo desarrollado constituye un prototipo y es prueba de que es posible aplicar la detección y reconocimiento de objetos para determinar la presencia de violencia; específicamente armas de fuego, en un contenido multimedia.

- **Generar una ontología de armas de fuego siguiendo una metodología específica.**

Se realizó una investigación sobre las metodologías para creación de ontologías que podrían servir para la representación de armas de fuego, llegando a seleccionar finalmente la metodología NeOn. El uso de esta metodología fue clave debido a que mantiene una secuencia de pasos ordenada y concreta, llevando a generar finalmente un modelo ontológico mediante la reutilización de ontologías afines

a la temática tratada. Se encontró además que existe solamente una ontología conceptual sobre la clasificación de armas de fuego por lo que se tomó esta como base referente para la creación de la nueva ontología.

- **Fusionar la ontología de armas de fuego con una ontología para la modelación de programas de televisión dando como resultado una ontología que permita la anotación de armas de fuego en video.**

Para al cumplimiento de este objetivo se realizó una búsqueda de ontologías para la modelación de programación televisiva que puedan servir en el proceso de anotación semántica. Como resultado de esta búsqueda se obtuvo dos ontologías candidatas a las cuales se les aplicó un proceso de selección por calificación de parámetros. De dicha tarea se concluyó que Programmes Ontology se adaptaba de mejor manera y pudo ser fácilmente integrada con la ontología de armas de fuego. Previo a la integración se realizaron ciertas modificaciones sobre la ontología Programmes con la finalidad de que se pueda integrar correctamente en una sola ontología que permita realizar el proceso de anotación semántica de videos.

Para proyectos de índole similar, es recomendable aplicar inicialmente algún proceso de selección de ontologías mediante la calificación de parámetros utilizando pesos. Esto llevará a seleccionar la ontología más adecuada y que se pueda integrar fácilmente con otras ontologías o componentes involucrados en el proyecto.

- **Anotar semánticamente un video en base a los términos de la ontología para la posterior búsqueda y/o consulta de resultados y generación de los prototipos correspondientes.**

Fue necesario contar con una ontología que por medio de propiedades y entidades permita anotar las armas de fuego y los videos en los cuales se detectaron dichas armas. La ontología se obtuvo mediante un proceso de reutilización y vinculación de dos ontologías, tal como se describió anteriormente. El proceso de anotación se realizó mediante un módulo programado para dicho fin. Para el desarrollo del módulo se aplicó las librerías de Apache Jena que manejan de manera eficiente las ontologías en lenguaje Java. Manipular tanto la ontología de armas de fuego así como la Programmes de contenido televisivo aportó al mayor entendimiento por parte del grupo de trabajo en cuanto a este tipo de tecnologías y permitió comprobar su utilidad para modelar situaciones del mundo real.

Por último, se desarrolló un módulo de consulta que accede al servidor Virtuoso en donde se encuentran almacenadas las ontologías. Para ello se utilizaron

varias tecnologías web ampliamente conocidas como HTML, CSS, PHP, entre otras. Es generalmente el objetivo de un proyecto informático contar con resultados que sean presentados al usuario final, de ahí la importancia de esta fase del proyecto. Se recomienda poner especial énfasis en la satisfacción de los requerimientos funcionales esperados, pues de ello dependerá también la satisfacción del usuario. Es así que, desarrollar un módulo que responda a los cuestionamientos del usuario y además le permita interactuar directamente con el sistema es vital para el éxito de cualquier proyecto.

Cómo se puede observar, cada objetivo contribuye al cumplimiento del objetivo general del proyecto que es consolidar en un proceso establecido las tareas realizadas para la anotación de objetos en video, de manera que constituya un documento de referencia útil para desarrolladores que se encuentren implicados en proyectos afines al descrito en este documento.

6.2. Futuras Líneas de Investigación

A la culminación de un proyecto, es común pensar en modificaciones o nuevos requerimientos que se puedan desarrollar con la finalidad de enriquecer el resultado obtenido previamente. A continuación se describen algunas líneas de investigación que pudiesen abordarse en un futuro para contribuir con el desarrollo de la propuesta presentada a lo largo de este documento.

- **Optimización del algoritmo de detección y reconocimiento de objetos**

Debido a que el algoritmo actual presenta limitaciones como que el tiempo requerido para el análisis de un video es considerablemente alto en relación a la duración real del video, así como la orientación de los objetos constituye un factor que limita la capacidad del sistema para reconocer las armas de fuego y por otro lado a medida que el dataset crece el tiempo de respuesta se incrementa, se plantea realizar un estudio más exhaustivo de los algoritmos que pudieran contribuir a contrarrestar estas falencias.

El punto de partida para dicha tarea iniciaría con la investigación acerca de la segmentación semántica de imágenes; que es un campo dentro del procesamiento de imágenes el cual propone dar características propias a los píxeles y luego analizarlos para determinar así áreas, y por consiguiente objetos. Esto contribuiría a un reconocimiento más eficiente, pues se basaría en las características propias



de los objetos, permitiendo también prescindir de la necesidad de un dataset que contenga las imágenes a ser reconocidas.

- **Análisis del audio contenido en el video**

A pesar de que el análisis de video basado en su contenido visual permite alcanzar un reconocimiento bastante acertado de objetos, el hecho de determinar si una escena es violenta o no, puede resultar ambiguo dependiendo de otros factores presentes en el video. Para lograr que el sistema actual sea más eficaz, se propone realizar un análisis del audio contenido en el video, puesto que existen factores auditivos determinantes de violencia tales como lenguaje violento explícito, sonidos de golpes, disparos, explosiones, etc. Esto complementaría el actual sistema para brindar resultados más acertados.

- **Vinculación del proyecto con un sistema de recomendación de contenido televisivo**

Contando con resultados que determinan la presencia de violencia en el contenido televisivo se presenta la oportunidad de vincular lo propuesto en este proyecto con un sistema de recomendación de contenido televisivo, que tome en cuenta tanto las características del usuario así como sus preferencias en cuanto al contenido que desea ver en televisión.

Anexos



UNIVERSIDAD DE CUENCA
desde 1867

Anexo A

Configuración de Variables de Entorno para OpenCV

A continuación se describe el procedimiento para la configuración de las variables de entorno necesarias para el adecuado funcionamiento de OpenCV.

En primer lugar se debe descargar el paquete de librerías de OpenCV desde el siguiente enlace oficial: <http://opencv.org/downloads.html>.

Hecho esto se debe ejecutar el instalador y únicamente seguir el asistente de instalación.

Ahora, es necesario que se configuren ciertas variables de entorno con la finalidad de que el IDE QtCreator reconozca y pueda hacer uso a las librerías de OpenCV. Para ello se deben realizar los siguientes pasos:

1. Ingresar a *Sistema* y dar clic en *Configuración avanzada del sistema*.

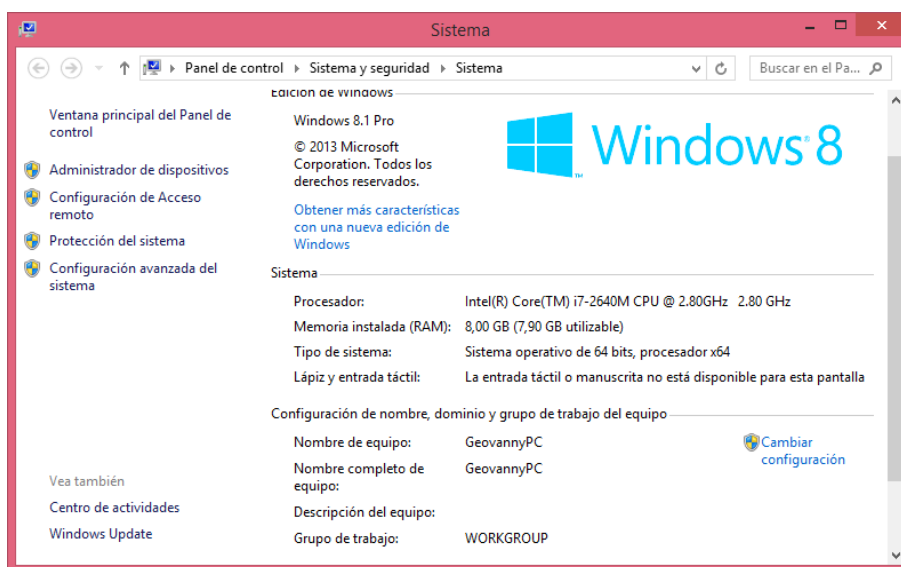


Figura A.1: Ventana de configuración avanzada del sistema.

2. La Figura A.2 muestra la ventana de *Propiedades del sistema* donde se debe seleccionar la pestaña *Opciones Avanzadas* y dar clic en el botón *Variables de Entorno*.

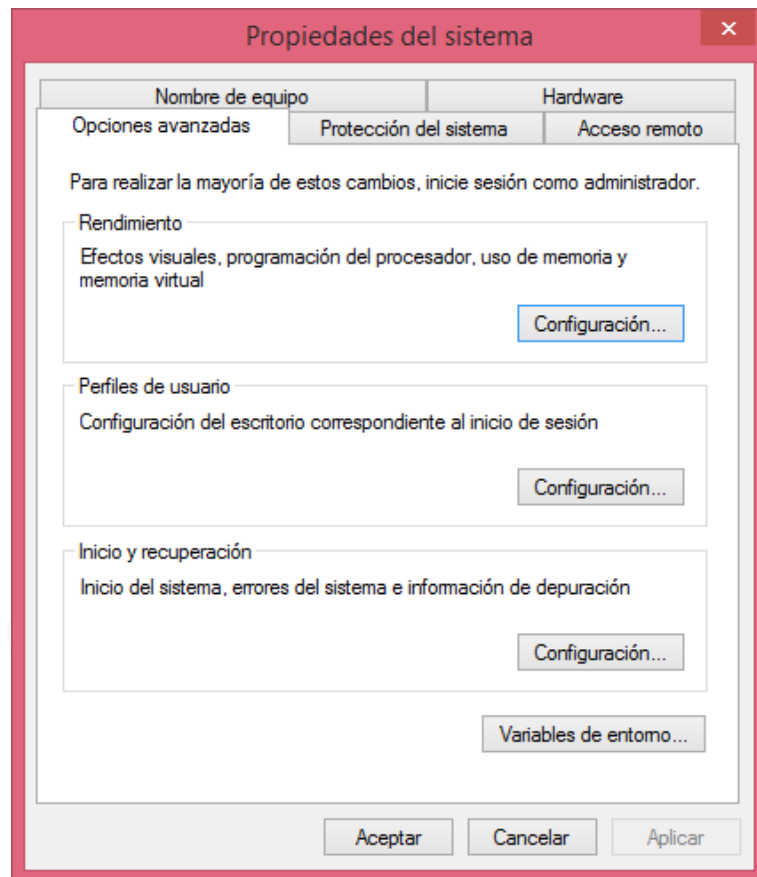


Figura A.2: Ventana de propiedades del sistema.

3. En la ventana de Variables de entorno, en la sección variables de sistema se deben agregar las variables *OPENCV_BUILD* Y *OPENCV_DIR*, como se muestra en la Figura A.3.
4. Hecho esto, clic en Aceptar para guardar los cambios. Es recomendable reiniciar el computador ya que existen ocasiones en las cuales las librerías no son reconocidas sino hasta el reinicio del sistema.

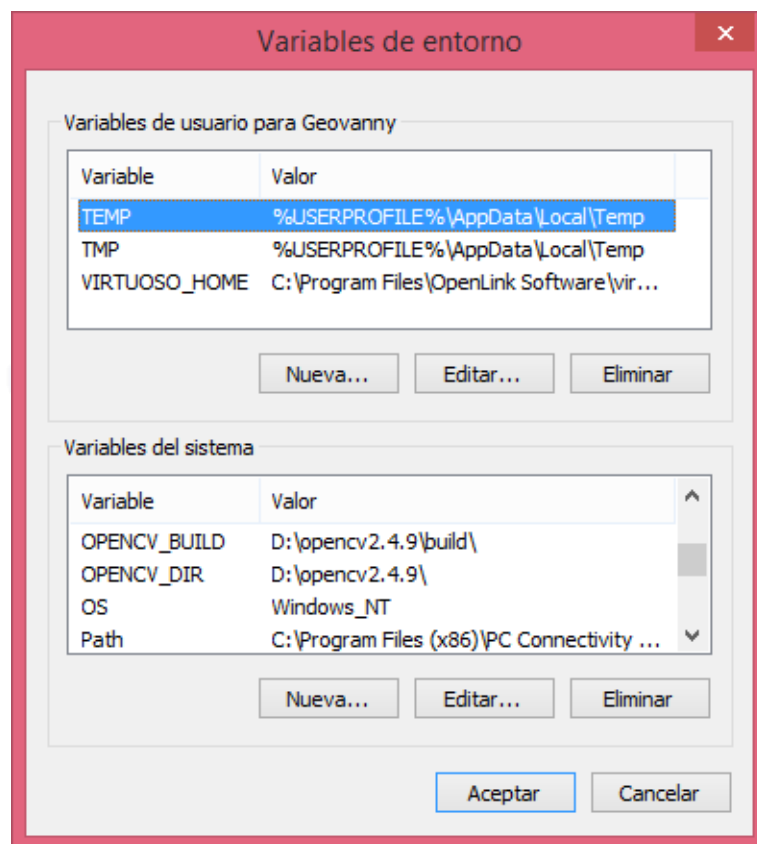


Figura A.3: Ventana de configuración de variables de entorno.

Anexo B

Ejecución del Módulo de Reconocimiento de Objetos

A continuación se describe brevemente el proceso de ejecución del módulo de reconocimiento de objetos en video. Este proceso se realizó varias veces durante el tiempo de pruebas para poder determinar la efectividad del módulo desarrollado.

Al ejecutar el módulo se presenta la interfaz de la Figura B.1, que consta de un campo para ingresar el directorio del cual se cargará el video. Tener en cuenta que dicho directorio debe ser el mismo referido en el campo “url” de los objetos almacenados en la base de datos de objetos modelo.

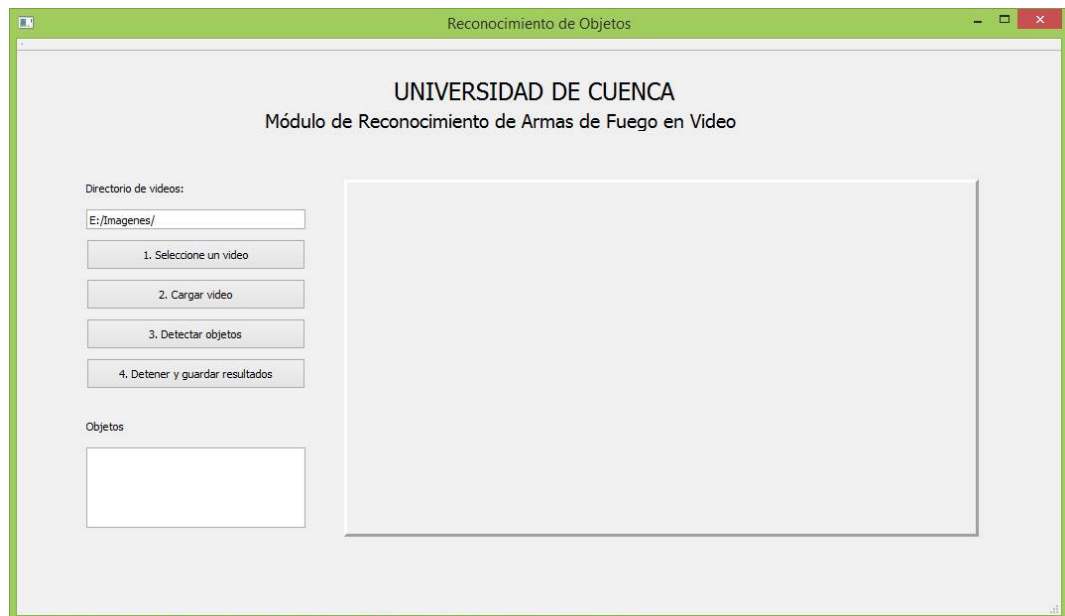


Figura B.1: Interfaz del módulo de reconocimiento de armas en video.

A continuación se describen los pasos necesarios para la ejecución del módulo.

1. Clic en el botón *Seleccione un video* y elegir el video a analizar.

2. Presionar el botón *Cargar video* para mostrar el video en pantalla. Como resultado el video se reproducirá como se observa en la Figura B.2.



Figura B.2: Imagen representativa de un video en reproducción.

3. Presionar *Detectar objetos* para iniciar el proceso de reconocimiento. Como se observa en la Figura B.3, el objeto reconocido aparecerá encerrado dentro de un borde verde, además en la parte inferior izquierda aparecerá el nombre del objeto y el tiempo de aparición con respecto al video, expresado en milisegundos.
4. Presionar el botón *Detener y guardar resultados* para almacenar los resultados del reconocimiento en el archivo .XML.

Para el ejemplo abordado se utilizó un video de 31 segundos de duración, aparecen tres armas en el mismo, a diferentes intervalos de tiempo. El tiempo requerido para procesar todo el video fue de 9 minutos, esto puede variar dependiendo de la calidad del video y de las dimensiones de las imágenes base, así como de la cantidad de objetos almacenados en la base de datos.

Se obtiene como salida un archivo en formato .XML. Como producto de la prueba llevada a cabo en esta ejecución de prueba, el contenido del archivo fue el que se muestra en el Segmento de código B.1.



Figura B.3: Imagen representativa del reconocimiento exitoso de tres objetos simultáneamente.

```

<Objetos>
<Objeto>
  <Identificador>1</Identificador>
  <Nombre>REVOLVER CO2 DAN WESSON 6“</Nombre>
  <Inicio>3966</Inicio>
  <Fin>9454</Fin>
  <IdVideo>8</IdVideo>
</Objeto>
<Objeto>
  <Identificador>1</Identificador>
  <Nombre>REVOLVER CO2 DAN WESSON 6“</Nombre>
  <Inicio>21785</Inicio>
  <Fin>28958</Fin>
  <IdVideo>8</IdVideo>
</Objeto>
<Objeto>
  <Identificador>2</Identificador>
  <Nombre>PISTOLA BERETTA 92 FS</Nombre>
  <Inicio>4892</Inicio>
  <Fin>9553</Fin>
  <IdVideo>8</IdVideo>

```



```
</Objeto>
<Objeto>
  <Identificador>2</Identificador>
  <Nombre>PISTOLA BERETTA 92 FS</Nombre>
  <Inicio>21553</Inicio>
  <Fin>28628</Fin>
  <IdVideo>8</IdVideo>
</Objeto>
<Objeto>
  <Identificador>3</Identificador>
  <Nombre>ESCOPETA MOSSBERG 590 MARINER</Nombre>
  <Inicio>8793</Inicio>
  <Fin>16264</Fin>
  <IdVideo>8</IdVideo>
</Objeto>
<Objeto>
  <Identificador>3</Identificador>
  <Nombre>ESCOPETA MOSSBERG 590 MARINER</Nombre>
  <Inicio>21619</Inicio>
  <Fin>28264</Fin>
  <IdVideo>8</IdVideo>
</Objeto>
</Objetos>
```

Segmento de código B.1: Contenido del archivo .xml obtenido como salida.

Como se puede observar, aparecen los tres objetos reconocidos y los respectivos intervalos de aparición, así como el video en el cual se reconoció, cumpliendo con el objetivo específico que plantea realizar el reconocimiento de armas de fuego en video. Estos datos se utilizan en el Capítulo V para el proceso de anotación semántica de armas de fuego en video.

ACRÓNIMOS

- AFA** *Algoritmo de Análisis de Forma*. 30
- API** *Application Programming Interface*. 70
- BSD** *Berkeley Software Distribution*. 52
- CSS** *Cascading Style Sheets*. 84
- FLANN** *Fast Library for Approximate Nearest Neighbors*. 35, 45, 49, 53, 87
- HTML** *HyperText Markup Language*. 83, 84
- HTTP** *Hypertext Transfer Protocol*. 82, 84
- IDE** *Integrated Development Environment*. 51, 52, 70
- ISDB-T** *Integrated Services Digital Broadcasting - Terrestrial*. 18
- JPEG** *Joint Photographic Experts Group*. 34
- JSON** *JavaScript Object Notation*. 82–84
- MPEG-2** *Moving Pictures Experts Group 2*. 18
- MPEG-4** *Moving Pictures Experts Group 4*. 18
- NOR** *Recursos No Ontológicos*. 58
- ODP** *Patrones de Diseño de Ontologías*. 63
- OR** *Recursos Ontológicos*. 58
- OWL** *Ontology Web Language*. 38, 60, 70

- PHP** *PHP Hypertext Pre-processor*. 83, 84
- PLN** *Procesamiento del Lenguaje Natural*. 40
- PNG** *Portable Network Graphics*. 34
- RAE** *Real Academia de la Lengua Española*. 24, 25, 40
- RDF** *Resource Description Framework*. 38, 69–71, 79, 80
- SGBD** *Sistema Gestor de Base de Datos*. 39, 52, 84
- SIFT** *Scale-Invariant Feature Transform*. 32
- SPARQL** *SPARQL Protocol and RDF Query Language*. 39, 70, 71, 82–84, 86
- SURF** *Speeded Up Robust Features*. 32–34, 44, 45, 49, 53, 87
- URI** *Uniform Resource Identifier*. 68, 70
- URL** *Uniform Resource Locator*. 50
- W3C** *World Wide Web Consortium*. 38
- XML** *eXtensible Markup Language*. 38, 58, 70, 78, 79, 87, 88

Bibliografía

- [1] M. Espinoza and K. Palacio, “Empleo de tecnologías semánticas para el análisis de contenido multimedia transmitido para televisión digital terrestre,” *Departamento de Ciencias de la Computación - Universidad de Cuenca*, 2014.
- [2] L. E. Sucar and G. Gómez, “Visión computacional,” *Instituto Nacional de Astrofísica, Óptica y Electrónica, Puebla, México*, 2011.
- [3] Open Source Computer Vision, “Meanshift and camshift,” 2015, [En línea]. Disponible en: http://docs.opencv.org/master/db/df8/tutorial_py_meanshift.html.
- [4] Z. Shang, “Bayesian variable selection: Theory and applications,” Ph.D. dissertation, University of Wisconsin–Madison, 2011.
- [5] M. Watson, A. Bhatti, H. Abdi, and S. Nahavandi, *Generation of 3D sparse feature models using multiple stereo views*. InTech, 2012.
- [6] Open Source Python Tutorials, “Feature matching,” 2015, [En línea]. Disponible en: http://opencv-python-tutroals.readthedocs.org/en/latest/py_tutorials/py_feature2d/py_matcher/py_matcher.html.
- [7] M. C. Suárez-Figueroa, A. Gómez-Pérez, E. Motta, and A. Gangemi, *Ontology engineering in a networked world*. Springer Science & Business Media, 2012.
- [8] Sony Corporation, “The beginning of the digital age,” 2015, [En línea]. Disponible en: <http://www.sony.net/SonyInfo/CorporateInfo/History/SonyHistory/2-04.html#block4>.
- [9] J. W. Woods, *Multidimensional signal, image, and video processing and coding*. Academic press, 2011.

- [10] Asamblea Nacional del Ecuador, “Ley orgánica de comunicación,” 2013, [En línea]. Disponible en: http://www.asambleanacional.gob.ec/es/system/files/ley_organica_comunicacion.pdf.
- [11] S. B. Suárez, “Biblioteca semántica de webquest,” Ph.D. dissertation, Universidad de Valladolid, 2004, [En línea]. Disponible en: <http://www.infor.uva.es/~sblanco/Tesis/Anotaciones%20Sem%C3%A1nticas.pdf>.
- [12] L. Alegsa, *Diccionario de Informática y Tecnología*. ALEGSA, 1998, [En línea]. Disponible en: <http://www.alegsa.com.ar/Dic/imagen.php>.
- [13] Á. Martínez-Nistal, “Porcesamiento digital de imagenes,” *Servicio de Proceso de Imágenes y Tecnologías Multimedia - Universidad de Oviedo*, 2014.
- [14] R. C. Gonzalez and R. E. Woods, *Digital Image Processing (2nd Edition)*. Pearson, 2002.
- [15] T. León Quintanar, “Sistemas expertos y sus aplicaciones,” 2007.
- [16] A. N. Arslan, N. M. Sirakov, and S. Attardo, “Weapon ontology annotation using boundary describing sequences,” in *Image Analysis and Interpretation (SSIAI), 2012 IEEE Southwest Symposium on*. IEEE, 2012, pp. 101–104.
- [17] A. Moore, “An introductory tutorial on kd-trees. extract from phd,” Ph.D. dissertation, Thesis, Tech. Report, 1991.
- [18] S. Zheng, M.-M. Cheng, J. Warrell, P. Sturgess, V. Vineet, C. Rother, and P. H. Torr, “Dense semantic image segmentation with objects and attributes,” in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*. IEEE, 2014, pp. 3214–3221.
- [19] P. Arbeláez, B. Hariharan, C. Gu, S. Gupta, L. Bourdev, and J. Malik, “Semantic segmentation using regions and parts,” in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 3378–3385.
- [20] E. Causa, “Algoritmos para el análisis de formas y reconocimiento de patrones bitonales,” *Proyecto BIOPUS*, 2008.
- [21] D. H. Ballard and M. J. Swain, “Indexing via color histograms,” *IEEE Transactions on Image Processing*, pág, pp. 390–393, 1990.

- [22] M. Herigert, N. Bouchet, D. Peralta, R. Romani, and I. Pianetti, "Reconocimiento de imágenes mediante scale invariant feature transformation (sift)," 2010, [En línea]. Disponible en: http://www.frsf.utn.edu.ar/cneisi2010/archivos/04-Reconocimiento_de_Imagenes_SIFT.pdf.
- [23] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.
- [24] A. Haar, "On the theory of orthogonal function systems," *Math. Ann*, vol. 69, pp. 331–371, 1910.
- [25] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration." *VISAPP (1)*, vol. 2, 2009.
- [26] C. G. Cambronero and I. G. Moreno, "Algoritmos de aprendizaje: knn & kmeans," *Inteligencia en Redes de Comunicación, Universidad Carlos III de Madrid*, 2006.
- [27] I. Polikoff and D. Allemang, "Top quadrant technology briefing, semantic technology version 1.1," 2003.
- [28] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowledge acquisition*, vol. 5, no. 2, pp. 199–220, 1993.
- [29] N. Guarino, *Formal ontology in information systems: Proceedings of the first international conference (FOIS'98), June 6-8, Trento, Italy*. IOS press, 1998, vol. 46.
- [30] J. Tauberer, "What is rdf and what is it good for," *Retrieved March*, vol. 25, p. 2012, 2008.
- [31] W. W. W. Consortium *et al.*, "Rdf vocabulary description language 1.0: Rdf schema," <http://www.w3.org/TR/rdf-schema/>, 2004.
- [32] R. T. Pereira and Y. J. M. Aular, "Los lenguajes de representación semántica y su uso en la construcción de ontologías," *Revista de Ciencias Sociales*, vol. 13, no. 1, 2007.
- [33] O. W. Group, *Web Ontology Language (OWL)*, World Wide Web Consortium, [En línea]. Disponible en: <http://www.w3.org/2001/sw/wiki/OWL>.

- [34] D. Janssen and J. Cory, *Semantic Repository*. Technopedia, 2015, [En línea]. Disponible en: <http://www.techopedia.com/definition/28198/semantic-repository>.
- [35] W. W. W. Consortium *et al.*, “Sparql lenguaje de consulta para rdf,” 2008.
- [36] J. A. G. Luna, M. L. Bonilla, and I. D. Torres, “Metodologías y métodos para la construcción de ontologías,” *Scientia et Technica*, vol. 2, no. 50, pp. 133–140, 2012.
- [37] G. Toro del Valle, “Avanzando hacia la web semántica a través de la anotación de contenidos,” *Cuadernos de Comunicación e Innovación*, no. 1, 2011.
- [38] S. R. Román Jarrín and J. J. Sáenz Peñafiel, “Modelamiento del perfil de un usuario utilizando tecnologías semánticas,” Bachelor Thesis, Universidad de Cuenca, Agosto 2014.
- [39] B. M. Villazón-Terrazas, “A method for reusing and re-engineering non-ontological resources for building ontologies,” Ph.D. dissertation, Universidad Politécnica de Madrid, 2011.
- [40] M. Grüninger and M. S. Fox, “Methodology for the design and evaluation of ontologies,” 1995.
- [41] M. Espinoza and V. Saquicela, “Modelando los hábitos de consumo televisivo usando tecnologías semánticas,” 2013.
- [42] M. C. Suárez-Figueroa, “Neon methodology for building ontology networks: specification, scheduling and reuse,” Ph.D. dissertation, Informatica, 2010.
- [43] RENAR, *Clasificación de las armas de fuego*, Ministerio de Justicia y Derechos Humanos, [En línea]. Disponible en: http://www.renar.gov.ar/index_seccion.php?seccion=clasi_armas_fuego.
- [44] W. E. Winkler, “String comparator metrics and enhanced decision rules in the fellegi-sunter model of record linkage.” 1990.